INTEGRATED SCHEDULING FOR AMBULANCES AND AMBULANCE CREWS

Claire Elspeth Reeves

Bachelor of Science
(Physics and Mathematics) with Honours
University of Queensland

Principal Supervisor: Professor Erhan Kozan

Associate Supervisor: Professor Vo Anh

Submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy (Mathematics)

Statistics and Operations Research Discipline

Mathematical Sciences School

Science and Engineering Faculty

Queensland University of Technology

July 2015

Keywords

Ambulances, Ant Colony Optimisation, Constructive Heuristics, Disjunctive Graph, Dispatching, Flexible Flow Shop, Heuristics, Hybrid Heuristics, Scheduling, Tabu Search.

Abstract

Ambulance and ambulance crews operate in a complex, budget constrained environment where demand is expected to increase. This thesis addresses questions surrounding the optimisation of ambulance services through the development of Integer Programming (IP) models, utilising Flexible Flow Shop Scheduling (FFSS) techniques, for integration of ambulance scheduling and ambulance crew scheduling. Shift scheduling rules are included as constraints in a model that schedules the processing of each incident on available ambulances.

Three models are developed and tested using realistic data based on incident data provided by Queensland Ambulance Services (QAS). The first model is static, allowing ambulances to be dispatched from only one location each, and is tested using deterministic data. The second model allows dynamic relocation and reassignment of ambulances during a shift. It is tested with deterministic data using a rolling horizon approach; the results show a reduction in the number of ambulance crew shifts required compared to the static model. The third model is a real time model which searches for the best ambulance assignments and locations each time a change in the system occurs. Overtime is considered in dynamic and real time models through innovative use of disjunctive constraints that compel a job that is introduced to return an ambulance to the appropriate station to be scheduled at the end of every ambulance shift. The real time model is planned to be of use as a decision aid tool for ambulance dispatchers.

The FFSS formulations for the integrated scheduling models are NP-hard. The number of cooperating ambulances and facilities in a metropolitan region leads to a large number of decision nodes. Heuristic algorithms, based on the extended disjunctive graph, are developed to solve large problems. Promising results are being obtained from Constructive Heuristics (CH), Tabu Search (TS), Ant Colony Optimisation (ACO) and hybrid heuristics.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACO	Ant Colony Optimisation
BW	Bandwidth
CH	Constructive Heuristic
DARP	Dial a Ride Problem
ED	Emergency Department
EDD	Earliest Due Date
EMS	Emergency Medical Services
FCFS	First Come First Served
FFSS	Flexible Flow Shop Scheduling
FIFO	First In First Out (same as FCFS)
FJSS	Flexible Job Shop Scheduling
GA	Genetic Algorithm
HM	Harmony Memory
HMCR	Harmony Memory Consideration Rate
HS	Harmony Search
IP	Integer Programming
JSS	Job Shop Scheduling
LP	Linear Program
LPT	Longest Processing Time
MILP	Mixed Integer Linear Program
MINLP	Mixed Integer Non Linear Program
MIP	Mixed Integer Programming

OFV	Objective Function Value
OR	Operations Research
PAR	Pitch Adjustment Rate
PSO	Particle Swarm Optimisation
QAS	Queensland Ambulance Services
SA	Simulated Annealing
SPT	Shortest Processing Time
TL	Tabu List
TS	Tabu Search
VNS	Variable Neighbourhood Search
WAH	Weighted Ambulance Hours

Glossary

This glossary contains terms relevant to the ambulance environment and Job Shop Scheduling.

Ambulance	The vehicle that must be staffed by an ambulance crew in order to be active
Ambulance crew	A fixed mixture of staff that may be scheduled to work in an ambulance, subject to rules surrounding crew schedules
Ambulance scheduling	The process of allocating jobs to an ambulance crew staffing an ambulance
Arrival time	The time at which an ambulance arrives at an assigned destination
Bypass	occurs when an overstressed emergency department redirects all arriving ambulances to another hospital
Clear time	The time at which an ambulance completes all tasks for an assigned job
Coverage	The area that can be reached within a specific time window by available ambulances
Crew schedule	The set of shifts allocated to an ambulance crew
Dispatch time	The time at which an ambulance is assigned to a job and commences a response
Due date	The time at which a task is required to be completed
Feasible solution	A solution to the mathematical model that satisfies all of the constraints
Incident	An event requiring a response from an ambulance crew at a specific location
Job	A task or set of tasks that must be scheduled on some set of machines

Machine	A unit that is able to process a job or task (e.g. an ambulance staffed by an ambulance crew)
Makespan	The time elapsed from beginning a job until completion (i.e. for dispatch time to clear time)
nth Percentile	The percentage of instances that met or exceeded a given target
Priority	The category into which an incident requesting emergency medical services is triaged
Processing time	The amount of time required to complete a task (e.g. time to travel between two points)
Ramping	The situation where ambulances are forced to wait at a hospital before being able to transfer their patient to the emergency department; may also be referred to as offload delay
Reassignment	The process of re-allocating an ambulance already travelling to one task to a different task; may also be referred to as pre-emption
Release time	The time when a job becomes available
Relocation	The process of assigning an available ambulance to travel to a new location to wait for jobs and improve coverage; may also be referred to as redeployment
Response time	The time elapsed between the either the notification of an incident and the time that an ambulance arrived at the scene
Shift Scheduling	The process of allocating resources (e.g. appropriate ambulance crews) onto a fixed set of possible shifts
Tardiness	The amount of time elapsed after a due date until a task is completed

Statement of Original Authorship

The work contained in this thesis has not been previously submitted to meet requirements for an award at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

QUT Verified Signature

Signature:

Date: 30/07/2015

Acknowledgements

I would like to acknowledge the following people for their suggestions and support while working on this thesis. Firstly, I wish to express my gratitude to my principle supervisor, Professor Erhan Kozan, for feedback and direction over the last three years and acknowledge my associate supervisor, Professor Vo Anh. I would also like to thank the members of the panels for my confirmation and final seminars: Professor Glen Tian, Professor Vivienne Tippett, Professor Vo Anh, Associate Professor Paul Corry and Dr Robert Burdett. Professor Vivienne Tippett, in particular, offered valuable insights into health care systems.

I would like to thank the following people from Queensland Ambulance Services: Dr Emma Bosley, Deputy Commissioner David Eeles, Mr David Hill and Commissioner Russell Bowles and for their interest in this project and assistance in obtaining data.

I would also like to acknowledge the Queensland University of Technology for providing resources to enable me to undertake my research, and professional editor, Dr Christina Houen, for editing this thesis according to the guidelines of the Institute for Professional Editors (IPEd).

Thanks are due to my colleagues working in Operations Research at QUT for being available to discuss approaches to problems. Ruth Luscombe deserves a special thanks for the many discussions and emailed resources that have made the whole PhD process a lot easier and enjoyable.

I would like to show my appreciation for the support of my parents, both for encouraging me to start the PhD and being willing to support me in day to day activities so that I could focus my energies on my studies. My friends at Jugger merit a mention for being part of the sport and community that provided me with a much needed outlet and escape from the stresses of a PhD. Finally, to my wonderful partner Brendon, thank you for listening to me when I needed to talk about my problems, encouraging me in my work and making life easier through the long days towards the end of this project.

Chapter 1: Introduction

The problem explored in this thesis is the development of an integrated scheduling model for ambulances and ambulance crews. This is important due to the need to manage vital, yet costly services.

In this chapter, Section 1.1 details a problem statement establishing an understanding of the environment in which ambulance services operate. This understanding allows the objectives and scope of the project to be defined in Section 1.2. The research aims of this thesis are introduced in Section 1.3 and the remainder of the thesis is outlined in Section 1.4.

1.1 PROBLEM STATEMENT

This research project develops mathematical models to aid in the scheduling of ambulances and ambulance crew. Ambulances operate in a budget constrained environment where demand is expected to grow. Optimising the use of resources should clarify the minimum resources required to meet current and future demand, so that service can either be provided within budget, or a strong case presented for further service funding. The full problem covered in this project involves: analysing time-dependent demand; development of a strategic planning model to optimise an ambulance crew schedule that meets expected time-dependent demand; development of a real time scheduling model to optimise resources in near real-time; and, developing and testing solution algorithms for the scheduling models.

Ambulance services provide emergency medical service to people in a complex environment with several competing objectives. Service should be provided as quickly as possible in order to optimise patient outcomes; however, resources are often limited by cost constraints and availability. Ambulance services are expensive to run and staffing costs are the major cost for running ambulance services, comprising around 70% of the ambulance budget (Queensland Department of Community Safety, 2012; Queensland Treasury, 2007). The problem is further complicated by the possibility of the existence of multiple types of responder vehicles, which may require different staff and incur different costs to run. Objective functions that are applicable in this environment include timeliness objectives

(minimise time to any call, maximise area number of calls that can be met within a threshold) and objectives based on minimising cost or balancing workload (Goldberg, 2004).

Demand for ambulance services is on an upward trend, fuelled by an increase in urgent and life-threatening emergency incidents, which make up just over 70% of the total incidents to which ambulance services in Queensland respond. Of these, approximately 85% of incidents will be transported to a hospital (Queensland Ambulance Service, 2012). The decision to transport a patient to a particular hospital is dependent on several factors. For the cases where transportation is necessary, ambulance services face a choice of suitable hospitals, where preference for different hospitals may exist but not always be binding. Cases where a patient requires treatment at a hospital with a specialist unit attached, such as a cardiac care or major trauma unit, or where a patient is suffering post-op complications and should attend the facility where the original operation took place, are potentially more binding constraints than patient preference. Other factors such as the closest facility or the status of the emergency department are also considered.

A strategic approach to the problem attempts to determine the best locations to place limited resources in, in order to ensure a certain level of coverage, or, with a fixed set of resources, to maximise coverage. Coverage refers to the area that can be reached within a set time limit from the location of resources. The problem may include known patient transport that can be scheduled in advance; however, demand for emergency medical services is not able to be known exactly in advance. Historical data may be used to provide estimates of areas with high emergency demand and/or times when emergency demand is at a peak, in order to improve the model's optimisation of resource allocation.

There is also a dynamic component to ambulance services involving gaps in coverage that occur whenever an ambulance is dispatched to a call. Whenever an ambulance is busy, it is then unable to respond to a new call in the area that it would generally cover. Urgent ambulance calls must receive a response within a time window and the creation of a gap in coverage generates the situation where an ambulance from further away responds to a call for emergency medical assistance. Figure 1-1(a) provides an example of this situation for a simplified system. Station s_3 is the closest ambulance station to $d_1(t_1)$ (demand at destination d_1 at time t_1). It

dispatches an emergency medical response vehicle to $d_1(t_1)$. If no other ambulance is located at station s_3, this station is unable to respond to further calls until the current call has been cleared. In the event that a call at the location $d_3(t_3)$ arrives, where $t_3 <$ t_1 plus the amount of time taken to service the call from $d_1(t_1)$, then the closest station (s_3) is unavailable and an ambulance is instead dispatched from station s_1. This creates a much longer response time for $d_3(t_3)$. One method of dealing with this problem is to relocate and redeploy ambulances in real time. This is illustrated in Figure 1-1 (b). Station s_3 dispatches an ambulance to $d_1(t_1)$ and it is recognised that this creates a gap in coverage. If it is expected that the next call for an ambulance is more likely to come from the area around s_3 than s_1, or if station s_1 has spare resources, then an ambulance may be diverted from s_1 to s_3 at some point after time t_1. Demand hotspots, areas from where calls are expected to arise at a higher rate, can be estimated from historical data. This approach is used for expected coverage models.

A second way of dealing with the problem of gaps in coverage is to create new dispatching policies which can also be used with dynamic redeployment. Alternative dispatching policies, such as the example in Figure 1-1 (c), may also be created. In this example, an ambulance from the second closest station (s_1) is sent to the first job, as it will still be able to reach the call at $d_1(t_1)$ within response time limits but will leave a less significant gap in coverage than dispatching an ambulance from station s_3.

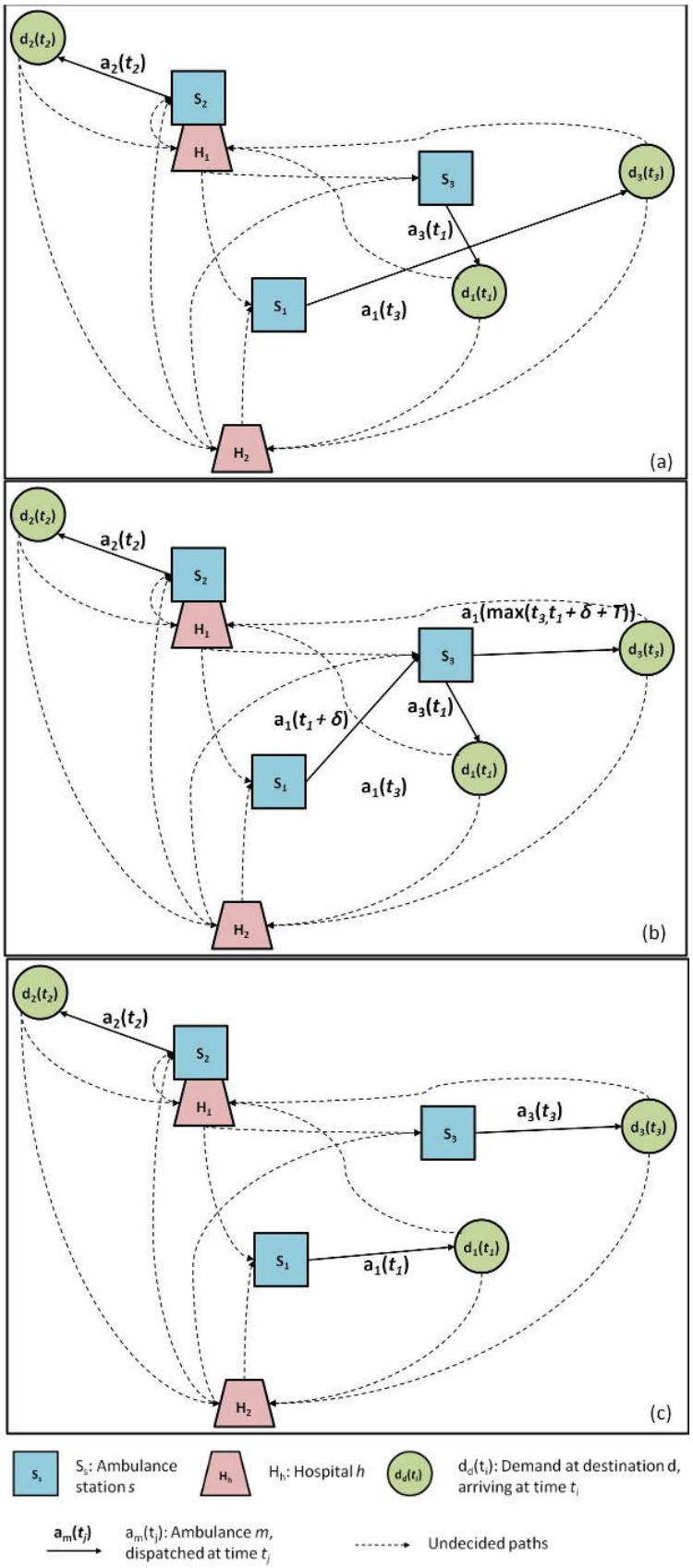


Figure 1-1 Example of dispatch rules for a simplified ambulance system where solid arcs represent paths travelled and dotted arcs represent possible decisions

1.2 SCOPE AND SIGNIFICANCE

The model for integrated ambulance scheduling and crew scheduling dispatches ambulances to meet demand during each shift while also considering the placement of ambulance crew onto shifts. Ambulance scheduling assigns calls for ambulance services (incidents) to individual ambulances staffed by an ambulance crew. Ambulance crew scheduling involves allocating crews to shifts in a manner that considers a given set of business rules. In this thesis, it is assumed that teams of staff are fixed into the correct staff mixture and scheduled together as a single crew. Different types of ambulances require different crews, which affects running costs.

Existing ambulance dispatching models fail to consider the effect of ambulance movements that may place ambulances far from their home station towards the end of a shift and create more overtime than is necessary. Overtime is a common occurrence in real life and increases the cost of running ambulance services. The models proposed in this thesis address this issue by considering an objective function that minimises the costs of both regular ambulance crew shifts and overtime. Overtime is of particular importance in metropolitan areas where ambulances are frequently relocated during a shift, and consideration should be given to the home station and shift end time for each ambulance when making decisions on dispatch or relocation.

Metropolitan and semi-urban areas, with more densely populated regions than remote and rural areas, have a higher call rate for requests for emergency medical services (EMS) and more complicated traffic conditions. These areas are of intense interest for optimising ambulance services. Not only are there a large number of incidents in these areas but there are larger numbers of ambulances, and multiple ambulance stations and hospitals which are able to co-operate to provide overlapping services across a large area.

The positive effect of this is a system that can utilise ambulances from one locality to respond to incidents in another and can relocate ambulances to provide area-wide coverage. Ambulances transporting patients to hospitals in metropolitan areas, where multiple hospitals are available, are able to play an important role in managing load share during times when emergency department capacity is stressed. This can feed back into lower ramping times, and thus lower overall time from dispatch to clear for ambulances.

The problem with this type of system is the additional complexity that is added by the increased number of possible options for scheduling ambulance relocations and dispatches. This complexity lends itself to mathematical analysis and optimisation. The research project applies scheduling techniques to help ambulance services in metropolitan areas minimise costs through optimising use of resources in a complex environment. The work is not expected to be applicable for extension to rural areas due to the different nature of demand.

Ambulance services are subject to a number of performance measures. The most used performance measure for ambulance services is the response time. McLay and Mayorga (2010) show that maximising the number of calls where an ambulance arrives on scene within response time thresholds can be used as an effective performance measure related to patient survival. The time window of the response time threshold depends on the severity of the call and is more important to meet for high priority emergencies. Other performance measures include the cardiac arrest survival rate and the average time from dispatch to clear.

The performance measure of response time thresholds will be considered in this research project as constraints. Other constraints in the model will reflect real life rules for ambulance crew scheduling. The business rules surrounding crew scheduling in the ambulance domain include, but are not limited to: the number of ordinary hours worked per week; maximum length of shift; scheduled days off; forward rotation of shifts; and restrictions on consecutive night shifts. The overall objective will be to minimise costs by minimising the number of ambulances required and overtime used.

There are ethical considerations relevant for EMS which impact on how constraints should be interpreted. A number of these considerations are discussed in Becker et al. (2013). Among the questions they ask are the following:

Is it ethically justifiable for EMS to deny or delay transport for a patient who does not have an emergent medical condition?

Is it ethically justifiable for EMS to deny a patient transport to a specific hospital at the patient's request?

and

What drives EMS personnel duty hours, and is there an optimal balance between performance and fatigue?

This thesis informs the first proposal by allowing non-emergency conditions to have reasonable delays, but no denial of service, if the overall system is closer to optimal. The second proposal is addressed by constraining the hospitals to which an ambulance may transport a patient. Patient preference may be modelled by restricting the number of allowable hospitals to which transportation may occur, and the effects on overall optimality and time spent travelling to, ramping at and admitting patients into hospitals investigated. The third question will be informed by how frequently overtime and disrupted meal breaks occur in optimised schedules, and by the utilisation rate of ambulances.

There are problems related to EMS that are worthy of note, but are considered out of scope for this research project. These include: mass casualty disaster events; ambulance deployment along highways or within rural areas; vehicle routing for EMS; and, emergency department (ED) management. Mass casualty disaster events require ambulances, but the objectives for these events may be different enough from objectives arising out of daily ambulance requirements that the models developed here are not guaranteed to be as effective for mass casualty disasters. While objectives for EMS in disasters include "maintaining normal services at an appropriate level", they may also require containment of the emergency and different approaches to treatment, including "promoting self-help and recovery" (Wilson, Hawe, Coates, & Crouch, 2013, p.644). Vehicle routing for emergency medical services, while important, is not within scope for this project. This thesis assumes that appropriate vehicle routing occurs prior to the models being solved, with estimated travel times for routes occurring as input. Emergency department management affects the ramping time experienced by ambulances. This thesis considers ramping time from historical information as modelling the interface between ambulance services and hospitals is beyond the scope of this project.

Attention has been drawn to the issues of ramping and hospital bypass in Queensland in recent times (Rosengren, 2012). Ramping refers to the situation where ambulances are forced to wait for a significant period of time before being able to transfer their patient to the hospital's emergency department. Bypass occurs when an overstressed emergency department redirects all arriving ambulances to another hospital. The effects of ramping and bypass are increased patient time in an ambulance and increased 'time to clear' for ambulance crews. Load sharing between

hospitals and recent directives to halt the practice of going on bypass have been proposed as methods to attempt to reduce the time spent waiting in an ambulances. Efficient and effective communication between Queensland Health and QAS have been recommended by Rosengren (2012) as a key to improving ramping figures. Ramping effects are considered as input for this thesis, not as an objective for minimisation.

1.3 RESEARCH AIMS

The **aims of the research** are as follows:

- Determine most utilised ambulance station locations and allocation of ambulances to each ambulance station for maximal coverage;
- Determine minimum number of ambulance crews required to meet demand under the fixed ambulance location condition and ambulance relocation condition;
- Determine an algorithm to recommend dispatching decisions using a minimal number of ambulance crews;
- Propose optimal ambulance crew scheduling methodology and validate;
- Include ambulance crew scheduling rules as constraints.

The proposed **research questions** are:

- Is a mathematical model with minimum simplifying assumptions for real-life ambulance dispatch, scheduling and crew scheduling necessary?
- What are the benefits of an integrated scheduling and crew scheduling model as opposed to a multistage model?
- What are the benefits, limitations and difficulties of a job shop scheduling approach to formulating the ambulance scheduling problem?
- What are the benefits of implementing a dynamic model rather than a static model?
- What are the benefits of a real time, dynamic scheduling approach (as opposed to a static, strategic planning model)?
- What algorithms are suitable for providing a timely, on-line solution? Is a new heuristic needed?

- What is the impact of the new solution technique (in terms of optimality and computational time required to solve)?
- What are the anticipated benefits and costs to ambulance services of the application of this methodology for a decision support tool?

1.4 THESIS OUTLINE

The remainder of this thesis is set out as follows: Chapter 2 presents a literature review discussing the different approaches that have been used to solve problems occurring within the environment of EMS; Chapter 3 presents the research outline for this project, including a discussion of the proposed methodology and solution techniques; Chapter 4 elaborates on the heuristic and metaheuristic solution techniques and discusses the suitability for the problem; Chapter 5 introduces the case study and analyses the data used to test the models developed for this research project; Chapter 6 presents the first model developed during this project, an integrated scheduling and crew scheduling model for ambulances under a static location dispatching condition, and contains the formulation, solution approach, results and analysis of the results; Chapter 7 presents the second model, which is a dynamic redeployment extension of the first model; Chapter 8 presents the third and final model, a real time model formulated to optimise ambulance assignments and locations, using scheduling techniques to be solved in near-real time using hybrid heuristic techniques; Chapter 9 summarises the three models, discusses how the results from this project may be applied, outlines how the proposed research aims have been met and the contribution made to the literature, and recommends further avenues of research.

Chapter 2: Literature Review

This chapter explores and appraises the literature related to optimising ambulance services. The topics reviewed include coverage, relocation and dispatching models (Section 2.1); scheduling models for emergency services (Section 2.2); and patient transportation models (Section 2.3). Related areas of interest are discussed in Section 2.4 and implications are presented in Section 2.5. A detailed summary of the literature reviewed in this chapter is shown in Table 2-1.

Table 2-1 Summary of literature surveyed on mathematical modelling with application to the emergency medical services environment

Section	Topic	References
2.1	Optimising ambulance resources	
2.1.1	Coverage models	(Brotcorne, Laporte, & Semet, 2003; Li, Zhao, Zhu, & Wyatt, 2011)
2.1.1.1	Deterministic coverage models	(Church & ReVelle, 1974; Toregas, Swain, ReVelle, & Bergman, 1971)
2.1.1.2	Expected coverage models	(Beraldi & Bruni, 2009; Daskin, 1982; Ingolfsson, Budge, & Erkut, 2008; Repede & Bernardo, 1994)
2.1.1.3	Dynamic coverage models	(Rajagopalan, Saydam, & Xiao, 2008)
2.1.1.4	Hypercube models	(Geroliminis, Karlaftis, & Skabardonis, 2009; Iannoni & Morabito, 2007; Larson, 1974; Mendonça & Morabito, 2001)
2.1.2	Relocation models	(Andersson & Värbrand, 2007; Gendreau, Laporte, & Semet, 2006; Goodwin & Medioli, 2013; Ibri, Drias, & Nourelfath, 2010; Maxwell et al., 2014; Maxwell, Restrepo, Henderson, & Topaloglu, 2010; Schmid & Doerner, 2010; Zhang, 2012)
2.1.3	General assignment models	(Haghani & Yang, 2007; Maleki, Majlesinasab, & Sepehri, 2014; Yang, Hamedi, & Haghani, 2005)
2.1.4	Simulation & dispatching strategies	(Goldberg, 2004; Haghani, Tian, & Hu, 2004; Henderson & Mason, 2005; Henderson & Mason, 1999; Kozan & Mesken, 2005; Zhen, Sheng, Xie, & Wang, 2014)
2.2	Shift scheduling and rostering	(Ernst, Jiang, Krishnamoorthy, Owens, & Sier, 2004; Ernst, Jiang, Krishnamoorthy, & Sier, 2004; Pinedo, 2012)
2.2.1	EMS crew/shift scheduling	(Aubin, 1992; Erdogan, Erkut, Ingolfsson, & Laporte, 2010; Rajagopalan, Saydam, Sharer, & Setzler, 2011; Trudeau, Rousseau, Ferland, & Choquette, 1989)
2.2.2	EMS rostering	(Li & Kozan, 2009; Xiang, Yin, & Lim, 2014)
2.3	Patient Transportation Models	

2.3.1	Travel time estimation	(Mason, 2005; Westgate, Woodard, Matteson, & Henderson, 2013)
2.3.2	Dial-A-Ride Problems	(Beaudry, Laporte, Melo, & Nickel, 2009; Kergosien, Lenté, Piton, & Billaut, 2011; Melachrinoudis, Ilhan, & Min, 2007; Melachrinoudis & Min, 2011; Parragh, 2009)
2.4	Related areas of application	(Almehdawe, Jewkes, & He, 2013; Azadeh, Hosseinabadi Farahani, Torabzadeh, & Baghersad, 2014; Yi & Kumar, 2007)
2.5	Summary	

2.1 OPTIMISING AMBULANCE RESOURCES

Due to the expense of running ambulance services and the potential consequences of inadequate provision of emergency medical attention (that is, poor patient outcomes in life-threatening situations), optimising ambulance services is an important problem. Literature on optimisation of ambulance resources dates back to the 1970s. A large amount of this literature addresses the topic of coverage, mostly through Integer Programming (IP) formulations, although dynamic programming, queuing theory and simulation also have a place in the literature.

2.1.1 Coverage Models

Solutions from coverage models indicate the lower limit on the number of ambulance vehicles and crews required to be in service at all hours of the day. The total number of resources and location of facilities needed to meet demand may be solved individually as a set-location coverage problem (Church & ReVelle, 1974; Gendreau, Laporte, & Semet, 1997; Ingolfsson, et al., 2008; Toregas, et al., 1971). Modifications to the coverage problem allowing dynamic relocation of resources have also been proposed (Andersson & Värbrand, 2007; Gendreau, et al., 2006; Schmid & Doerner, 2010).

Brotcorne, et al. (2003) and Li, et al. (2011) provide summaries of methods used to solve the coverage problem. Brotcorne, et al. (2003) describe several IP formulations and how they apply to the ambulance problem. Li, et al. (2011) take this further and describe some heuristic solution methods for IP models. For large scale problems, heuristic algorithms such as Genetic Algorithms, Tabu Search, Simulated Annealing and Ant Colony Optimization have been used to solve IP models and simulation used to test optimality.

2.1.1.1 Deterministic Coverage Models

Early ambulance coverage models were deterministic models focused on the strategic problem of locating ambulance facilities and determining the number of ambulances to base at each facility. Church and ReVelle (1974) and Toregas, et al. (1971) deal with the static problem using IP to find the minimum number of resources to cover all demand and maximise the coverage for a given number of resources. The model formulated in Toregas, et al. (1971) is simple enough to be solved with standard linear programming codes. Church and ReVelle (1974) test solutions for the IP model found through greedy heuristics, linear programming and the branch and bound exact method. The heuristics are non-optimal but are close when compared to solutions by the other methods. The main issue with these early models is that they assume that once a facility is located, it is available at all times. This does not represent the real life situation because once an ambulance has been dispatched it is unavailable to answer another call. Daskin (1982) substantiates the claim that the deterministic models overestimated coverage and many later models introduce stochastic elements to incorporate a busy probability for ambulances, leading to estimates of expected coverage.

2.1.1.2 Expected Coverage Models

Expected coverage models improve static coverage models through estimations of the fraction of time that ambulances are busy (Beraldi & Bruni, 2009; Daskin, 1982; Ingolfsson, et al., 2008; Repede & Bernardo, 1994). The expected coverage model concept is introduced in Daskin (1982). Similar to the deterministic coverage model, the locations at which ambulances are positioned are optimised in order to maximise coverage. The difference is that expected coverage considers the probability that ambulances in certain positions will be busy and, hence, unavailable to respond to calls. Queuing theory is used to work out the percentage of time ambulances will be unavailable to respond to calls. Daskin (1982) shows, with two test problems, that expected coverage decreases with higher probability of an ambulance being busy, but there exists a range of busy probability values for which a solution of ambulance locations remains optimal. They show that the more locations at which an ambulance may be positioned, the better the overall coverage for both deterministic and stochastic coverage.

Repede and Bernardo (1994) develop an expected coverage model to strategically position ambulances, extended to allow for time dependent demand. They compare results from their model, using simulation, with a stochastic formulation that does not include time dependent demand when estimating ambulance availability. The total number of ambulances required to maintain coverage experiences improved. Reductions achieved in terms of response time are slight when compared with deterministic coverage models but significant compared to best practices at the time. Both Daskin (1982) and Repede and Bernardo (1994) assume that the busy probability of each ambulance is independent. This is not the case in reality (as ambulances being unavailable in an area will increase the chance of ambulances nearby being dispatched) but is still able to provide insight for strategic planning. More recent literature addresses this assumption.

Ingolfsson, et al. (2008) extend the expected coverage models to include random delays prior to commencing travel and random travel times when estimating ambulance availability. They also address the problem of dependent ambulance availability through including the probability that an ambulance will be dispatched from the i^{th} preferred location. Results from Ingolfsson, et al. (2008) provide evidence that deterministic delay and average travel times underestimate the number of ambulances required and overestimate coverage. The more variation there is in delays, the more significant this effect.

Beraldi and Bruni (2009) deal with dependence between busy ambulances through a two-stage coverage model. One stage locates ambulances while the other, instead of using queuing theory to determine expected availability, optimises reliability; that is, how often ambulances from the preferred location are available to be dispatched. They are able to show the cost impact of requiring higher reliability levels.

2.1.1.3 Dynamic Coverage Models

Dynamic coverage models seek to maximise coverage and/or minimise the number of ambulances required for a time-dependent coverage problem. This requires them to be solved repeatedly at time intervals, resulting in the best positions for ambulance at different times of the day. Dynamic coverage models are useful for planning shift schedules.

A dynamic coverage location model is presented in Rajagopalan, et al. (2008). Here, they require the minimum number of ambulances at the best locations that will meet coverage requirements to a given reliability level (percentage that the requirement will be met). A 'look ahead' procedure replaces expected coverage from a hypercube model to reduce computational effort. The model is solved across contiguous three hour time intervals using a Tabu Search algorithm. Rajagopalan, et al. (2008) predict fleet sizes to meet coverage requirements with 90% reliability for each time interval.

2.1.1.4 Hypercube Models

Several of the expected coverage models mentioned have been supplemented by queuing theory in the form of the hypercube model (Li, et al., 2011). The hypercube model was introduced in Larson (1974) as another approach to resource allocation problems for emergency services that is able to distinguish individual units. In terms of ambulance services, ambulances may be dispatched from designated zones, with preference to dispatch an ambulance from the same zone as an emergent incident or the closest zone if all preferred ambulances are busy. The state transitions allowed in the model are from an ambulance being idle to an ambulance becoming busy. The steady state is able to return information on the number of incidents receiving ambulances dispatched from outside the preferred zone and average travel times. Geroliminis, et al. (2009) expand the hypercube model idea and link it to the location problem. They use a hybrid formulation to minimise mean response time and maintain adequate coverage. This is a computationally complex optimisation model that requires heuristics to find solutions. Their results were compared with results from basic models maximising coverage or minimising response time and found that the proposed model offers improvements in response time when demand is high.

The hypercube model has also been applied to ambulances providing emergency medical services (EMS) along highways (Geroliminis, et al., 2009; Iannoni & Morabito, 2007; Mendonça & Morabito, 2001). These models divide a highway into a set of geographical atoms which can generate requests for EMS, allowing demand to have a spatial and temporal component. Ambulances are assigned to regions along the highway and atoms are allocated ambulances from which a response is preferred. If all preferred ambulances are busy, the next closest

will be selected or the incident lost to the system. The model presented in Mendonça and Morabito (2001) requires estimates of travel times and service times and allows ambulances to be in one of two states: available or busy. Ambulances always return to their base after responding to an incident. The equilibrium state can be analysed to investigate performance measures such as the number of times a backup ambulance was sent instead of the preferred ambulance, the number of lost incidents or average travel time while responding to an incident. Iannoni and Morabito (2007) describe a hypercube model for ambulances on highways that includes different types of EMS vehicles and incidents where double or triple ambulances are required. They show that the hypercube model can adequately represent EMS on highways and that the queuing model can be used to investigate performance measures from other optimisation models.

Hypercube models are limited by assumptions that must be made and by analysis only being possible on steady stage systems (Henderson & Mason, 2005). Henderson and Mason (1999) model the dispatch process of ambulances as an M/G/1queuing theory model in order to get an initial approximation of the number of ambulances required at available ambulance stations. They make the assumption that ambulance stations cannot send relief vehicles to nearby overstressed stations in order to simplify the model. Their preliminary results on a case study indicate a need for additional resources and a simulation model is developed to verify and refine the result with further relaxed assumptions.

2.1.2 Relocation Models

Relocation models (also referred to as redeployment models) have a fixed number of resources and are solved to maximise coverage at trigger events, indicating the best new positions for ambulances when the status of the system is altered. Relocation models are used to provide reactive decision support and are required to be solved in real time, or else look up tables are produced in advance to indicate relocations that will improve a system with a particular status.

Gendreau, et al. (2006) formulate a maximal expected coverage problem and solve it as a relocation problem for a small number of physicians' cars providing a medical response service similar to ambulances but without further transport to hospitals. The small number of vehicles in this case allowed the model to be solved

directly using IBM ILOG CPLEX Studios. Larger problem sizes can be tackled by the use of look-up tables, where the solutions for scenarios have been found in advance.

Andersson and Värbrand (2007) present a relocation model that will be resolved when the coverage level for any node drops lower than the desired number of ambulances. The objective of their model is to minimise the maximum travel time for the relocations necessary to ensure the area is fully covered. The risk of this approach is that scenarios may result in no solution as the required coverage level is unable to be met with the available resources. This is shown to occur more frequently at higher incident arrival rates. A heuristic solution approach is required that will always return feasible solutions. The results from Andersson and Värbrand (2007), evaluated with simulation, show that the relocation model improves the percentage of jobs met in response time limits, particularly where coverage limits are higher and more relocations occur.

Schmid and Doerner (2010) present a double coverage MIP formulation for a relocation model and solve it with local search heuristics. They consider time varying coverage, variations in travel time and solve a multi-period model optimising system wide coverage simultaneously. They show that time dependent variations in travel time are necessary during peak times because use of averages can result in the objective being overestimated. Both Andersson and Värbrand (2007) and Schmid and Doerner (2010) confirm that allowing additional locations as destinations for ambulance relocation improves response times and coverage.

Ibri, et al. (2010) formulate a mathematical model to integrate the problem of dispatching vehicles with the coverage problem. Ambulances are assigned to subsets denoting their current activity. Multiple objectives are proposed to apply penalties for lack of coverage, for not satisfying incidents in the required time, or for pre-empting assignments. They test the effectiveness of a hybrid heuristic (Ant Colony Optimization and Tabu Search) to find solutions.

Zhang (2012) presents a more in-depth exploration of relocation through investigation of dynamic programming, ranking tables and IP models, evaluated through simulations. Several models are presented, each increasing in complexity, based around the concept of moving up ambulances to the next closest location. Move up policies outperform static policies for ambulance location. However,

according to Zhang (2012) and Andersson and Värbrand (2007), excess relocation activity may be frustrating to crews.

Maxwell, et al. (2010) present an approximate dynamic program to solve ambulance redeployment decisions and measure the percentage of unsatisfied incidents under conditions of uncertain service and travel times. Decisions in the model are only permitted to occur at trigger events, i.e. incident arrivals and completion of ambulance assignments. Ambulances can only be redeployed immediately upon becoming available as a way to reduce relocations and may only be assigned to an ambulance base. The results from this model indicate that scenarios with congested resources have very little opportunity to relocate while scenarios with excess ambulance receive very little benefit from relocating. There is improvement to be made in the number of incidents reached from relocation for non-congested scenarios that still have limited resources.

Goodwin and Medioli (2013) explore ambulance redeployment, describing the set of ambulance deployment decisions that can be made as real-time control decisions, to minimise average response time for high priority incidents. This concept leads to a formulation of the ambulance scheduling problem as a stochastic Model Predictive Control problem and the development of an approximate dynamic programming algorithm to solve the model. Dispatch decisions and ambulances becoming available again act as trigger points for decisions. They are able to improve average response time for the 50th percentile of jobs.

Maxwell, et al. (2014) continue investigation of redeployment and seek to place lower bounds on the percentage of jobs expected to be tardy over a long period of time, from any redeployment policy, using a queuing model. The bound is intended to be used to determine if a schedule for ambulance services is adequate. The model uses a lower service time than expected in reality and optimally locating ambulances without required travel time when calls are received. In this way, the queuing model will always have at least as many ambulances available in the system as would be expected for a real life scenario. Simulation tests showed that the predicted bound on the percentage of tardy jobs is still much lower than the result from best deployment strategies for at least one of the realistic scenarios investigated. Further work suggested in the paper, on reducing the assumptions in the

model determining the bound and on improving redeployment policies, is necessary to demonstrate a more meaningful bound.

2.1.3 General Assignment Models

General Assignment Problems assign jobs to multiple agents, without exceeding the capacity of any one agent (Öncan, 2007). The general assignment approach to real-time emergency response fleet deployment is considered in-depth by Yang, et al. (2005) and Haghani and Yang (2007). They propose an IP mathematical model to solve the objective of minimising total weighted travel time. Higher priority emergencies will have a higher priority. This model considers different emergency vehicle types (including ambulances) for a set of vehicles in the system. At each time *t* in the model each vehicle is also part of one subset. These subsets include: vehicles idle at home base; vehicles moving towards an emergency; vehicles servicing an emergency on site; vehicles transporting from the emergency site to a hospital; vehicles remaining at a hospital; and vehicles travelling toward a station. Defining these subsets allows the model to let vehicles be relocated from hospitals and between stations at any time *t* while not in service, and to be re-routed to higher priority calls arising before the vehicle has reached the site of an emergency, or re-routed to a different hospital while transporting a patient. A coverage rate parameter is used to help determine if vehicles should be re-deployed to alternative ambulance stations.

General assignment using IP is a promising, though NP-hard approach for optimising ambulance deployment strategies. It allows demand points to be assigned to a specific subset of resources. The models in Haghani and Yang (2007) and Yang, et al. (2005) solve a general assignment mathematical model for a small problem using exact solution techniques for a timeliness objective. However, these models fail to take into account offload delays at hospitals due to the utilisation of the Emergency Department and subsequent effects on emergency vehicle availability. Their models place hospitals on an equivalent of bypass once capacity has been reached. This model also does not consider minimising resources or scheduling shifts of ambulance crews. Some exact solution techniques have been developed for mathematical programming models. In other cases, the problems are NP-hard and heuristic techniques are developed in order to find a good solution in a reasonable amount of time.

Maleki, et al. (2014) introduce a generalised assignment model to investigate redeployment of ambulances. In this paper, travel time is minimised. They are able to solve their relocation model to improve coverage when compared against existing policies using the same number of ambulances and reducing the percentage or incidents receiving tardy responses. However, they lack heuristic solution approaches and, as such, are restricted to problem sizes which can be solved by CPLEX. They also assume a homogeneous fleet of ambulances.

2.1.4 Simulation and Dispatching Strategies

Haghani, et al. (2004) use a simulation model to test various dispatching strategies for emergency service vehicles, while Kozan and Mesken (2005) explore simulation of the emergency call centre environment. Their model tests the resources required to handle dynamic demand in the call centre making the decision about which ambulance to dispatch. Goldberg (2004) discusses other literature on simulation models for Emergency Medical Services.

Henderson and Mason (1999) developed a simulation and analysis tool called BARTSIM to verify and refine results from a preliminary queueing model into ambulance resource location. This is used to determine the number of ambulance vehicles required at each station at different times of day and across the week. Henderson and Mason (2005) further discuss the simulation and visualisation package for ambulance services. Their model is a discrete event simulation model that uses real, recorded data rather than generated data and has a sophisticated model for estimating travel times. Heuristics are needed to solve the optimisation model for estimating travel times. Simulation results are presented on a geographic information system display to aid decision makers.

Zhen, et al. (2014) investigate the effect of three decision strategies for scheduling ambulances: intuitive scheduling, where the closest ambulance is assigned; region coverage scheduling, where only urgent cases are guaranteed the closest ambulance, and the ambulance selected for non-urgent cases should arrive within the time window and have the smallest effect on reducing total coverage; and a centrality-based approach to scheduling, where dispatching an ambulance is balanced against the importance of keeping the ambulance free at its location. Performance measures of average response time and, importantly, the number of tardy responses are compared for each of these dispatching strategies on stochastic

models. They confirm that coverage-based and centrality-based dispatching are able to reduce the percentage of tardy responses compared to intuitive, closest first scheduling. The centrality-based rule appears better under conditions where resources are tight, but further investigation and improvement of dispatching rules should be performed.

2.2 SHIFT SCHEDULING AND ROSTERING FOR AMBULANCES

From Pinedo (2012, p. 1) the definition of scheduling is that it "is a decision making process" that "deals with the allocation of resources to tasks over a given time period". Optimised results from the dynamic coverage problem define the resources required to meet time-dependent demand, which then provides input for the shift scheduling and rostering problems.

Definitions of shift scheduling and rostering from Ernst, Jiang, Krishnamoorthy, Owens, et al. (2004, pp. 21,27) are as follows: "Shift scheduling involves selecting a set of the best shifts from a (large) pool of candidate shifts on a single day" (p. 27); while the rostering problem involves "allocating suitably qualified staff to meet a time dependent demand for different services while observing industrial workplace agreements and attempting to satisfy individual work preferences."

The shift scheduling and rostering problem for emergency medical services are not as well examined in the literature as is the coverage problem. Ernst, Jiang, Krishnamoorthy, and Sier (2004) review staff scheduling and rostering processes for several areas of application, including emergency services such as ambulance services. However, while the scheduling and rostering problems appear in many fields, the nature of the demand for emergency medical services distinguishes the approach needed here from approaches used in other areas of application (such as air crew scheduling or nurse rostering).

2.2.1 EMS Crew and Shift Scheduling

Shift scheduling for emergency services is distinguished from other areas of application by the fact that demand is dynamic and not known ahead of time (Ernst, Jiang, Krishnamoorthy, & Sier, 2004). Demand has both a spatial and temporal component and forecasting plays an important role in scheduling emergency

services. Solving the coverage problem is one way of turning forecast demand into server requirements at a given spatial node at each point in time.

Trudeau, et al. (1989) explore the application of operations research techniques to ambulance scheduling, and show that a mathematical model better allocates emergency services to fit the demand profile than does manual allocation. Aubin (1992) extended this work to create a strategic shift schedule for ambulance services where demand for ambulance services had spatial and temporal characteristics and multiple priority types. A least-cost workday is found by solving a set-coverage model using linear programming or branch and bound techniques. Weekly schedules are developed by looking for cycles in workday schedules. They note that the increasing the degree of temporal resolution (e.g. solving for each minute as opposed to each hour) increases the complexity of the problem. Implementation of their procedure demonstrated monetary savings on ambulance running costs and improved the homogeneity of shift starting times.

The approach used in Erdogan, et al. (2010) schedules ambulance shifts for maximum coverage and is formulated as a two stage process. The first stage determines the static allocation of ambulances for maximum expected coverage and uses a Tabu Search algorithm to allocate ambulances to ambulance stations. A weekly ambulance crew scheduling model is proposed as a second stage to maximise service coverage each hour with the number of ambulances from the first stage and scheduling constraints.

A model presented in Rajagopalan, et al. (2011) also uses a two-stage integrated approach for ambulance deployment and shift scheduling. The first stage uses a dynamic expected coverage model to determine the number and location of ambulances needed at each two hour time period for each day of the week. This requires metaheuristics to be solvable in a reasonable amount of time. The solution for this stage is evaluated through the quality of the solution and the computation time to find it. The results from the first stage became input for the second stage, along with a selection of shift options. In the second stage, an IP model is solved to find a crew schedule that minimises the number of shifts. The fleet size and shift schedule obtained from the second model are tested through a simulation process to verify that coverage is met, or nearly met, for all times.

2.2.2 EMS Rostering

Li and Kozan (2009) present a two-stage mathematical programming model for rostering with the aim of maximising coverage and minimising crew. The first stage determined the starting times of each shift and the number of staff required to work that shift (that is, to create a shift schedule that satisfies demand). The second stage allocated staff to the schedule with the objective of minimising the total number of staff required. They propose the idea of an integrated mathematical model to solve the problem in a single stage model; however, this would become an NP-hard problem, particularly if staff personal preferences are included.

Also within the healthcare management environment, but not an emergency medical services solution, is the approach used by Xiang, et al. (2014) to integrate scheduling and rostering for operating room (OR) scheduling with nurse rostering constraints such as role and availability. The model looks at elective surgery on a daily scale and includes nurse rostering constraints within the model to schedule surgery and resources. Using an Ant Colony Optimisation metaheuristic, they are able to solve a schedule that balances resources better than current practice in a test case and reduces the total completion time. They apply their proposed methodology to another test case from the literature and show a reduction in nurse overtime, and variation in utilisation of operating theatres and total completion time. However, the time taken to run the heuristic to obtain a good solution depends on the size of the problem, and it may take several hours to plan two days of schedules.

2.3 PATIENT TRANSPORTATION MODELS

A key role of ambulances is to provide patient transportation services to hospitals, in addition to first response to emergency medical situations. These jobs may be known and planned in advance or be a result of dynamic emergency demand. This section reviews literature related to developing road networks for ambulance services and scheduling and routing of patient transportation provided by ambulances.

2.3.1 Estimating Travel Times for EMS

The ambulance problem includes vehicles capable of using lights and sirens to aid their passing through traffic and the ability to exceed speed limits for very severe incidents. Therefore, it may be desirable to create a unique model for

estimating travel time. An optimisation model for travel time would need to create a road network specific to the area of operation. This would consist of a set of nodes representing intersections and directed arcs representing the physical paths between nodes. Travel along any individual arc is assumed to be at a constant speed.

Henderson and Mason (2005) develop a road network model to provide data for a simulation model specific to their case study. To decrease the solving time required while the simulation model is running, some shortest paths are pre-determined for different times of the day. Mason (2005) and Westgate, et al. (2013) take advantage of automatic vehicle location tracking in emergency medical vehicles and look at 'map matching' to estimate the travel time for ambulances. Ambulance trips begin and end at nodes and travel along directed arcs. Recorded data from prior trips is used to extract estimates for the travel time along these arcs. The travel time of new trips can be predicted by determining their path and the travel time along each segment making up that path. This allows the effects of ambulance emergency lights and sirens to be considered when estimating travel time.

Westgate, et al. (2013) use a Bayesian model and allow for small errors in GPS data from recorded ambulance trips. A Markov Chain Monte Carlo method is used to sample the model and, for any new trip, they estimate the probability of a path being chosen and determine the most probable travel time along that path. Their tests show that their model for estimating ambulance travel times outperformed models based on local methods that estimate travel speed along arcs directly from mapped speeds. They also note that, as the size of the road network increases, the amount of computational power needed to solve for the path and travel time will also increase.

2.3.2 Dial-A-Ride Problems

Road networks with nodes and directed arcs are also a key feature of problems formulated as 'dial-a-ride' problems. The dial-a-ride problem (DARP) is of interest because it involves sending transportation from a depot in response to a call at another location and involves time windows specifying when a vehicle should arrive. DARP combines scheduling and routing to balance competing objectives and is different from other pickup and delivery problems as it has a focus on reducing user inconvenience In the case of emergency medical services, inconvenience to a patient could be considered to be a function of delayed response time and extended

time in transit and offloading of a patient to an emergency department. The dynamic DARP allows for demand to arise dynamically rather than the static case where all demand is known in advance. Dynamic DARP methods have been applied to transportation within the healthcare management environment (Beaudry, et al., 2009; Kergosien, et al., 2011; Melachrinoudis, et al., 2007; Melachrinoudis & Min, 2011).

Situations involving outpatients requiring transportation to a health organisation for treatment are considered in the literature (Beaudry, et al., 2009; Melachrinoudis, et al., 2007; Melachrinoudis & Min, 2011). These DARP formulations have soft time windows where patients are considered inconvenienced if transport arrives outside of these time windows. Outpatient transportation problems differ from the ambulance problem in that the destination of the transportation (i.e. a specific treatment centre) is known exactly at the time that a vehicle is sent to pick up a patient and transportation is able to be arranged in advance. The need to arrive at the site of a patient within a certain time frame is also relaxed to a soft constraint. Patients may also share transportation in the DARP problem.

Beaudry, et al. (2009) solve a dynamic transportation problem for patients in large hospitals using a heuristic procedure applied to a modified DARP. They make the assumptions that the shortest path between any two points is known and will be followed, and that vehicles cannot be redeployed once on route. Transportation within a hospital includes moving patients within buildings using wheelchairs, stretchers or beds and, in some campus style hospitals, between buildings using ambulance vehicles. Delay in transportation can have follow-on costs, as high cost specialised medical equipment may be idle while waiting for patients to arrive, or patients may experience delays with appointments because of a delay in transportation. Similar to the ambulance problem, the patient requiring transport has an earliest time at which pick-up can occur and a latest time after which pick-up may still occur but a penalty applies. Instant pick-up may be specified for patients, suggesting that the model proposed in Beaudry, et al. (2009) has similarities to the emergency medical response problem. Additional similarities are that dispatching decisions are made continually throughout the day and different types of vehicles and multiple depots are considered.

Melachrinoudis and Min (2011) formulate a healthcare dial-a-ride problem to find routes and schedules that minimise cost and user inconvenience. Their model

deals with outpatients who call for transport to reach healthcare services. The DARP for these paratransit services has time windows, precedence constraints and a combinatorial nature. Dynamic programming has been applied as an exact solution approach for DARP, but is limited in the problem sizes which it can solve. Melachrinoudis and Min (2011) solve their model using Tabu Search heuristics for problems of realistic size. They also tested a branch and bound algorithm for an exact solution but found it is unable to obtain good solutions in a practical amount of time. The Tabu Search heuristic finds solutions as good as or close to the known exact solutions for the problem sizes tested but with a greatly reduced solve time.

Kergosien, et al. (2011) also introduce a Tabu Search heuristic for patient transportation, allowing patients to have different priority types and request different types of ambulance vehicles. Patients are transported to facilities within a hospital or between hospitals. Rosters for ambulance crew are fixed in advance but staff may change vehicles several times during a shift as long as they return to a depot in order to do so. The heuristic attempts to find a feasible schedule of activities for each rostered ambulance crew. Activities known in advance are planned at the beginning of the day and the schedule updated as new activities become known. Their algorithm stores and updates feasible routes while the tabu list stores and compares objective values. The dynamic algorithm was tested for a case study and found to require fewer tasks to be subcontracted and none to be started outside the desired time window when compared to existing practices.

An in depth exploration of DARP for ambulance scheduling and routing may be found in Parragh (2009). The culmination of this work is a solution to a realistic problem for patient transportation that is able to meet both planned and dynamic demand effectively. The objective of the models presented by Parragh (2009) minimise total routing costs while the constraints ensure patients must appropriate resources in appropriate time windows. The requirement for an ambulance to return to an ambulance base at the beginning and ending of each shift in order to swap drivers is also modelled. Due to the complexity of the problem, computational times for real problems solved with heuristics are still significant.

2.4 LINKS TO EMERGENCY DEPARTMENT AND DISASTER RELIEF SCHEDULING

The literature reviewed in this section is presented for the insights which can be provided from scheduling problems in Emergency Departments (EDs), which experience similar demand profiles to ambulances and have similar requirements to treat patients according to triage priority and within an appropriate time window. The insights are also drawn from disaster relief scheduling, which requires the use of ambulances.

The main differences between ED scheduling and ambulance scheduling are that ambulances require a travel component to be considered within the problem, while emergency departments have more types of resources (e.g. doctors, nurses, medical equipment, operating rooms) affecting scheduling. Azadeh, et al. (2014) schedule patients in an emergency department and solve with a real case study. The aim is to minimise waiting time for patients as a function of priority. This is done by minimising makespan for each job, weighted by priority of job times. They use a flexible open job shop scheduling formulation, with patients as jobs and staff as machines. Open job shop is a generalisation of job shop scheduling where the sequence of operations is not fixed. They present a comprehensive formulation using disjunctive variables to prevent more than one task at a time and completion time variables constrained by processing times. The formulation is NP-hard, so a Genetic Algorithm was developed to solve the model such that it was able to return solutions to realistic problems in minutes. Tuning of Genetic Algorithm parameters is done via an experimental approach selecting independent parameters to vary and testing the response, and the algorithm is verified comparing several tests against branch and bound solutions. A case study shows the results from the proposed model and algorithm outperforming existing practices for total weighted makespan. Azadeh, et al. (2014) show metaheuristics for an NP-hard flexible job shop are able to solve realistic problems in reasonable time and return good solutions.

Improvements in ED scheduling may also improve the performance of ambulance services, because ambulances experience delays through offload delays at Emergency departments. Queuing theory is used to investigate ambulance offload delays, also known as ramping, which occurs when an ambulance is unable to transfer a patient into an Emergency Department (ED) immediately upon arrival at a hospital due to the facility being at, or over, capacity. Almehdawe, et al. (2013)

develop a Markovian queuing model to explore queue lengths and waiting times at the interface between ambulances and EDs. The model presented considers multiple hospitals able to receive walk-in patients and patients from ambulances, with each ambulance patient having a probability for assignment to hospitals. Patients in the queue are assigned beds as they become available, with preference, including the possibility of pre-emption, given to ambulance patients over walk-in patients. The number of ambulance patients in each ED, ambulances in offload delay at each ED, probability of all ambulances in offload delay, and walk-in patient queue parameters, are estimated at the steady state. Case studies, investigating effects of low/high in patient arrival rates, routing probabilities and service time rates, are tested alongside a realistic system. Almehdawe, et al. (2013) show prioritising ambulance patients over walk-in patients reduces offload delay but at the cost of extreme waiting times for walk-in patients. Using ED capacity to influence routing probability and balance ED utilisation decreases the expected number of ambulances in offload delay and the total offload delay. This is evidence that selecting appropriate hospitals for ambulances by including ED capacity in the decision making process is important for ambulances.

Major disaster relief problems use ambulance services to treat and move casualties and provide transportation to hospitals, with preference given to priority cases. However, the demand for these problems is much greater than for normal ambulance activity, with multiple casualties at the same locations exceeding capacity and requiring ambulances to re-visit locations. Yi and Kumar (2007) solve a mixed integer network flow model for disaster relief. This type of problem manages the flow of material between nodes, and Yi and Kumar (2007) specifically consider transportation of wounded people from demand nodes to hospital nodes. Response times and severity of wounds are taken into account. Disaster relief operations differ from daily ambulance operations, as transportation of multiple people in a single trip may be more common and several returns to a single node may be required. The authors find Ant Colony Optimisation (ACO) useful for speedy solutions, which is vital, because a planner in disaster situation needs to obtain and update solutions for routing and assignment quickly. An optimality gap exists but is small enough to be acceptable for the gains in runtime.

2.5 IMPLICATIONS AND SUMMARY

Stochastic coverage models are well represented in the literature, and improvements continue to be made to methods of estimating the steady state ambulance availability by including additional uncertainty.

Dynamic coverage models are useful to inform strategic shift scheduling models as the first stage of a two stage solution approach.

Relocation models, determining the best location of existing ambulance resources based on the current state of the system, are successful at reducing average response times and the percentage of jobs receiving responses outside the time window. It is difficult to solve these models, and questions remain about the effects of large numbers of relocations of ambulance staff. There is room in the literature for further improvement of heuristic solution techniques for relocation models to provide decision support in real time. General assignment models are a promising area for integrating ambulance scheduling and relocations; however, heuristic solution approaches to these models require further development and testing.

Dispatching strategies are explored through simulation models. These require potentially improving rules to be defined and, as such, cannot test any rule not explicitly considered.

Dynamic Dial-a-Ride Problems for ambulance scheduling are able to model emergency and pre-planned demand for ambulance transportation with time windows. Increasingly complex models are being developed that minimise user inconvenience and are able to introduce requirements for ambulance crew to begin and end shifts at ambulance depots. These too require heuristics to obtain solutions within useful time frames. Dynamic DARP formulations are an interesting approach to ambulance scheduling but have yet to be shown to be able to provide relocation decisions for improving coverage.

Heuristics shown to be effective in the literature are Tabu Search, Ant Colony Optimisation, Genetic Algorithms and Simulated Annealing, with multiple examples of successful hybrid heuristics. Location decisions from coverage and relocation models are often explored through simulation models and compared to existing ambulance deployment policies as a way to verify results from the models. Coverage, reliability for meeting coverage requirements, average response times and the number of tardy responses are the most used performance measures.

A gap in the literature has been identified as the formulation and solution of a single stage integrated shift scheduling and ambulance scheduling model. Furthermore, the problem of integrating relocation and dispatching policies is still largely unexplored, with plenty of room to reduce simplifying assumptions and trial new heuristic solution methods. This thesis will address the problems of integrating shift scheduling with ambulance scheduling to explore whether the expected number of ambulances to meet demand can be reduced. A new model for integrating relocation and dispatching decisions will also be developed and new heuristics developed for each model presented in the thesis.

A case study is used to test the novel models presented in this thesis. Direct assignment of ambulances to a realistic sample of incidents, as output from the scheduling model, will indicate performance on fleet size, response times and the number of tardy responses.

Chapter 3: Research Outline

This chapter proposes a topic of investigation for the thesis and outlines how the investigation is performed. Background information on Operations Research (OR), Mathematical Programming, Mixed Integer Programming (MIP) and Job Shop Scheduling (JSS) is presented**.** Three models are described in the solution approach to be verified with a case study based on ambulance demand in the Brisbane metropolitan area. The outputs of the models developed in this thesis include: ambulance crew shift schedules that ensure a sufficient number of ambulances are available at ambulance stations each shift while meeting rules from workplace agreements; and ambulance schedules, covering dispatch, hospital transfer and relocation activities.

In this chapter, Section 3.1 presents the research proposal; Section 3.2 presents necessary background on the techniques which will be applied in this thesis; Section 3.3 describes the methodology, including plans for model formulation and solution techniques; and Section 3.4 outlines the procedure which is followed for this thesis.

3.1 RESEARCH PROPOSAL

This thesis investigates the following proposals:

"A mathematical model integrating ambulance scheduling and ambulance crew shift scheduling can be formulated and solved using heuristic techniques such that a good solution is provided in a useful amount of time."

and

"A real time model for scheduling ambulances can be formulated and solved such that a good solution is provided in real time".

The first process to investigate these proposals is to develop a strategic model that can solve an integrated scheduling and crew scheduling problem for ambulance services. The model (or models) should have the minimum number of simplifying assumptions to accurately represent ambulance services. Upon development and satisfactory solution approaches for a strategic model (or models), a real time model is to be formulated that contains elements that allow information to be updated in the

model either every time information updates in the real world or at pre-defined time intervals. The real time model must use an ambulance crew shift schedule created by the strategic model as input, schedule incidents requiring ambulance services, and minimise disruptions to shift schedules from interrupted meal breaks and overtime.

Solutions to the models consider heuristic techniques. These are to be investigated to test the quality and speed of solutions that can be found for the strategic model/s and real time model.

3.2 BACKGROUND

This section introduces the discipline of Operations Research and the group of methods within the discipline that are essential to this thesis.

3.2.1 Operations Research

Operations Research (OR) is also known as Operational Research, Decision Science and even Management Science. It has its origins in World War II as a scientific process for exploring decision options, often using quantitative analysis. Modern OR still focuses on finding better solutions to complex decision making problems. Techniques applied to practical problems often involve the development of a model to represent the system described by the problem, followed by analysis of the solutions (Taha, 2003).

Operations Research is an interdisciplinary applied science involving techniques from mathematics, computer science and psychology. The definition of the problem is pivotal to the solution, and engagement with non-practitioners may be vital to obtaining a good outcome.

3.2.2 Mathematical Programming

Mathematical programming is an OR method for modelling and solving problems. Mathematical programming techniques, e.g. MIP, allow the real world problem to be abstracted into a quantitative form. A typical model consists of decision variables, an objective function and feasibility constraints. The objective function depends on the values that are taken by the decision variables and is either minimised or maximised through finding the best possible values for the variables.

Writing a problem as a mathematical program allows the complexity of the problem to be better understood and individual elements of the problem to be explored. This thesis aims to create multiple models for representing the real life situation faced by EMS with a minimal number of simplifying assumptions, and then develop and/or improve solution techniques to find good answers from these models in a reasonable amount of time.

The simplest version of mathematical programming is Linear Programming (LP). The objective and constraints are all linear functions of the decision variables, and decision variables may take on continuous values (Brucker & Knust, 2006). An Integer Program (IP) adds the condition that the decision variables can only take on integer values. If only some decision variables are required to be integer and others may be continuous, then the mathematical program is a Mixed Integer Program (MIP). The coverage problem described in the literature review is formulated using IP. Integer and Mixed Integer Programs are more difficult to solve than Linear Programs. For a LP, the optimal solution is found on the edge of the solution space. For an IP, the optimal solution at the edge of the solution space may not be integer and hence not a feasible solution. It becomes necessary to search within a larger section of the solution space to find an optimal integer solution.

Scheduling problems can be formulated using constraint programming techniques. This approach specifies interval decision variables for tasks that must be scheduled for processing on some set of resources. Disjunctive constraints are handled by specifying no overlapping of the intervals. Constraint programming solution techniques find a feasible solution that satisfies all constraints and then propagate additional precedence constraints to find an optimal schedule. This technique is suitable for makespan or due date objectives (Pinedo, 2012). It is proposed that the complex problem investigated in this thesis can use Job Shop Scheduling (JSS) techniques to formulate an appropriate model.

3.2.3 Job Shop Scheduling Problems

The classical Job Shop Scheduling problem has N jobs each consisting of a chain of n_j operations with some precedence constraints on the order of operations (Brucker, Jurisch, & Krämer, 1997; Brucker & Knust, 2006). There are m machines available to process operations, without pre-emption and limited to one operation per machine at a time. Operations have positive processing times and each operation

must be processed on a single machine that is part of some subset of the total set of machines. The objective of the classical JSS problem is to create a feasible schedule of jobs so that the minimum makespan is reached. A schedule is defined by starting times of all operations. A feasible schedule must obey all precedence relationships and prevent the overlapping of operations on a single machine.

For the classical JSS problem, minimising the critical path will minimise the makespan. Disjunctive graph models are a popular method used for JSS problems to find critical paths and determine the order of operations that are processed on the same machine. A disjunctive graph is a directed graph representing the possible schedules for all operations on all machines (Błażewicz, Pesch, & Sterna, 2000; Brucker & Knust, 2006; Pinedo, 2012). Each node on the graph represents an operation on a machine (including dummy operations *0* and *n+1* to represent a source and a sink). Nodes representing operations with fixed precedence constraints are connected by solid conjunctive arcs. Operations that require processing on the same machine that are not fixed in order obey the set of disjunctive constraints, which are represented by pairs of dashed lines on the graph. Arcs are weighted by the processing time of the node where the arc originates. Solving the disjunctive graph involves selecting one disjunctive arc from each pair of disjunctive constraints.

There are variations of the job shop problem that modify constraints to suit different assumptions. These include variations such as Flow Shop Scheduling (FSS), where all jobs follow the same order of processing but may be processed on different machines, and Flexible Job Shop Scheduling (FJSS), where there are multiple identical machines in parallel (Brucker & Knust, 2006; Pinedo, 2012). The flexible job shop scheduling problem (FJSP) is a generalisation of the classical job shop that requires assignment as well as sequencing. It is a strongly NP-hard problem as shown in Mati and Xie (2004). They demonstrate that even a two job FJSP is NP-hard for traditional objective criteria such as makespan and total tardiness. The relaxation of this assumption is highly important for the emergency medical response environment where demand is not known in advance. The dynamic FJSP relaxes the assumption that all jobs are known and ready for processing at time zero. Problems that are NP-hard generally require heuristic methods in order to obtain good solutions in a reasonable amount of time.

3.3 METHODOLOGY

This section discusses how the above mentioned techniques will be used to formulate a model specific to the integrated ambulance scheduling and shift scheduling problem that is investigated in this thesis. Solution techniques for this model are also proposed in this section.

3.3.1 Model Formulation

As specified above, JSS theory is used to formulate mathematical models to solve the research questions proposed in this thesis. The variation of the JSS problem that is of interest is Flexible Flow Shop Scheduling (FFSS). Each call for an ambulance may be regarded as a unique job where the machine processing the job is the crew of an ambulance. A novel formulation using disjunctive constraints and integer variables will be investigated to solve the problem as a flexible flow shop. This formulation approach is explored through the development of three mathematical models, discussed in Section 3.4.

The objective function for the strategic models is to minimise the total costs of running the ambulance services. This is done through minimising: i) total number of ambulance crew shifts, and ii) the amount of overtime worked. Tardy responses (i.e. calls for ambulance services not met within appropriate response time) are considered as a performance constraint rather than an objective. This approach sets these models in this thesis apart from many other approaches which use coverage requirements, response times and\or tardiness as objectives rather than enforcing them to meet certain requirements and minimising cost. The contribution to the literature from the strategic models is a methodology for building ambulance crew schedules from direct assignment of incidents to ambulances (and hospitals) so that ambulance crew schedules and ambulance schedules influence each other. This process also allows overtime to be measured, a factor that is a common occurrence for ambulance services but lacks investigation in the literature.

The ambulance crew shift schedule from strategic models is used as input for a real time model for scheduling ambulances. In the real time model, ambulance crews have a fixed shift schedule and the constraints meeting performance measures are relaxed. This model minimises tardiness, gaps in coverage, penalties for missed/interrupted breaks and overtime. Components of the objective may be investigated individually or be included in a weighted multiple criteria objective

function. This model and associated solution approach contribute to the literature on ambulance dispatch and relocation through the real time nature of the model, requiring fewer variables and the capability to solve quickly. Additionally, disjunctive variables allow i) a different method of estimating future ambulance availability, and ii) the ability to balance requirements that each scheduled shift for an ambulance should contain meal breaks and end the shift at the correct ambulance station with the entire schedule. As a result, emergency incidents are able to be scheduled alongside relocation events, meal breaks and jobs introduced to return each ambulance to its home ambulance station at the end of a scheduled shift with a reduction in unnecessary overtime.

3.3.2 NP-hardness

NP-hard problems are a set of problems that cannot be solved in polynomial time with any solution algorithm. Polynomial time means that the time to solve the algorithm depends on a polynomial function of the size of the problem. In other words, as the problem size increases, the solution time increases faster than a power of the problem size. For problems where the solution space increases exponentially as the number of decision variables *n* increases, a polynomial time algorithm is unlikely. Problem classes, in order of difficulty to solve, are:

- ***P*** Solutions can be found in polynomial time;
- ***NP*** A solution can be *verified* in polynomial time. NP stands for *nondeterministic polynomial time*;
- ***NP-hard*** The set of problems that are at least as hard to solve as the hardest problems in NP;
- ***NP-complete*** The set of NP problems that can be mapped to each other in polynomial time. The significance of this class is that if one problem is found to have a polynomial time solution, then all problems in this class will have a polynomial time solution.

Emergency vehicle and personnel scheduling problems have a complexity that can grow exponentially (Church, Sorensen, & Corrigan, 2001). The problem of scheduling ambulance services in this has a large number of interacting constraints and parameters that, considered together, make the problem complex. The ambulance scheduling problem consists of a set of jobs (i.e. calls for an ambulance)

to be scheduled onto multi-purpose machines (i.e. ambulances). All jobs have a set of operations (e.g. treating a patient, travelling to a hospital) that must be completed in a particular order. Each machine can only process one operation at a time. Operations are performed on one machine from a subset of all machines (because there are different types of ambulances). Processing time, for the ambulance problem presented in this thesis, depends on the machine selected for some operations but not all. Pre-emption is only permitted during certain operations. The inclusion of the ambulance crew scheduling adds a layer of complexity. Each machine can only be used at certain times and processing time in the ambulance scheduling problem varies according to decisions made in the ambulance crew scheduling component of the problem (i.e. ambulance station assignment affects travelling time). Integrating the two problems, using FFSS techniques, is NP-hard. Realistic scenarios are large enough that the NP-hard nature of the problem prevents exact solutions being found in reasonable time. Heuristic solution techniques are required to solve the models presented in this thesis.

3.3.3 Solution Techniques

Wang (2005) reviews solution techniques for FFSS problems. Methods exist for finding exact solutions to IP problems. If a disjunctive graph can be formulated for a scheduling problem with a minimal makespan objective then the solution can be found by minimising the critical path of the disjunctive graph. This involves selecting one disjunctive arc from each pair. The branch and bound technique is a simple yet powerful tool for solving IP models. The problem is solved as a linear programming problem and then branched at integer values closest to that solution for one parameter at a time. The solving and branching processes continue until an integer solution is found on one branch with an objective function that outperforms the current objective on every other branch. Scheduling problems solved using branch and bound are solved by branching along assignment of tasks to resources and following the branch with the lowest objective until the lowest objective is a complete solution. Branch and bound has been used to solve scheduling problems for flexible manufacturing under the makespan objective (Shanker & Modi, 1999). However, the complexity of the problem increases as the number of alternative resources for each task increases, and branch and bound becomes unwieldy for large scale flexible scheduling problems with many integer decision variables.

Commercial software, such as ILOG CPLEX Optimization Studios, exists and can be employed for solving suitable test cases of the scheduling problems. However, exact solutions are not able to be found for larger, more realistic problems.

Other possible solution methods considered include heuristics, hybrid heuristics and hyper heuristics. Heuristic methods are used to obtain good solutions to NP-hard problems within a suitable time frame by applying rules to search sections of the solution space. Metaheuristics (a sub-class of heuristics) are a class of computational methods of finding solutions to an optimisation problem where an exhaustive search of the solution space is impractical. These techniques attempt to optimise a performance measure (or fitness value) by iteratively trialling solutions.

Metaheuristics are better than randomly trialling solutions because they apply techniques to search the areas of solution space that are more likely to produce an optimal solution. The search technique can vary greatly depending on the measure given to optimise and the particular assumptions of the problem. Metaheuristic methods do not guarantee an exact optimal solution. Hybrid heuristics combine concepts from two heuristics, and can be used to create a two-stage algorithm for solving a problem.

Hyper heuristics use a learning process to select appropriate heuristics from a pre-defined set while searching the solution space. They are able to provide a general solution technique and are able to handle a larger variety of scenarios without the tuning of parameters required by metaheuristics and hybrid heuristics.

For this project, a real time model will require solutions that can be obtained within minutes. Heuristic techniques should be evaluated for optimality and solution speed. Heuristic methodologies are covered in greater detail in Chapter 4.

3.4 PROCEDURE

The first few months of this research study were spent researching the information needed to build a mathematical model with relevant constraints. This was facilitated by literature on ambulance services and optimisation and through liaison with QAS.

Three novel models were then formulated to investigate approaches to integrating ambulance scheduling and shift scheduling. Key assumptions from each model are shown in Table 3-1. Each model is briefly described as follows:

- The first model is the simplest. It uses deterministic data and dispatches ambulances from a static location. This model is relevant for strategic planning purposes.
- The second model expands upon the formulation of the first model by allowing ambulances to be relocated between stations during each shift. This is a more accurate representation of the system.
- The third model schedules ambulances in a similar fashion to the first and second models but attempts to optimise performance with available resources from a fixed shift schedule instead of optimising the shift schedule itself. The model is developed as a real time model which requires solution techniques able to find solutions in minutes or seconds.

Table 3-1 Components of each subsequent mathematical model

Model 1	**Model 2**	**Model 3**
Static station location for ambulance dispatch	Dynamic location for ambulance dispatch	
Deterministic data		Reactively gathered data

Each model is tested with a case study using EMS demand from Brisbane, Australia. Workforce Modelling reports containing the number of units working and incidents to which a response was sent per hour for each ambulance station were available as were more detailed incident logs containing the time at which key events occurred. These are analysed to extract information on shift patterns, dynamic demand profiles and expected processing times. A new set of data is then generated from the extracted parameters to provide the case study data. This data set is compared against the real data set to verify that it is a suitable representation of the real demand population.

Each model is then tested using the solution methods shown in Table 3-2. The resulting solutions are analysed to compare solution approaches and verify the models.

Table 3-2 Solution techniques applied to each model

Solution Technique	**Model 1**	**Model 2**	**Model 3**
CPLEX	✓	×	×
CH	✓	✓	✓

TS+CH	✓	✓	✓
ACO	×	✓	×
Hybrid ACO+CH	×	✓	✓

3.4.1 Model 1: Static Model with Deterministic Data

The first optimisation model that has been developed is a static and deterministic model that demonstrates that a Flexible Job Shop Scheduling Problem formulation is able to represent the ambulance problem. The decisions informed by the model are: how many ambulances of each type to assign to a shift; the ambulance station at which to place these ambulances; the weekly shift schedule for these ambulances (if the problem is solved for large enough scenarios); which ambulance, and when, to dispatch to each incident; the hospital to which a patient should be transferred; and, the amount of overtime to be expected each ambulance shift. The model minimises costs and the results provide an upper bound on the resources required at each station for each shift and how best to place them in order to reduce overtime. The details of this model are presented in Chapter 6.

3.4.2 Model 2: Dynamic Model with Deterministic Data

The second model extends the static model by allowing ambulances to be relocated to different ambulance stations when empty. Dynamically relocating ambulances reduces the number of ambulance vehicles required to meet demand. Overtime is determined by the clear times of new return-to-station jobs that are forced, by disjunctive constraints, to be the final job on each shift to which an ambulance is assigned. The model is solved using a rolling horizon, and reassignment of ambulances to different tasks is allowed each time a new horizon is solved. Further information on this model is located in Chapter 7.

3.4.3 Model 3: Real time Model

The third model is formulated to utilise real time information, and requires the ambulance shift schedule to be known when it is initialised. The objective is modified to reduce coverage and tardiness and may be adapted to include penalty costs for overtime and interrupted meal breaks. Ambulances can be dispatched from any location and reassignment of ambulances to different incidents or hospitals is allowed under certain conditions when new information becomes available. To return results in real time, the model would be required to be solved repeatedly. This

may be in reaction to data updating, whether through new jobs entering the system or information about current jobs being updated, or at designated time intervals. This requires a solution technique able to be solved and re-solved each time an ambulance is dispatched. A heuristic method is proposed that should enable the model to be solved within a suitable period of time (i.e. within minutes). Chapter 8 provides the details of the formulation and solution approach to the real time model.

3.4.4 Sensitivity Analysis

Heuristic algorithms for the static and dynamic models are run multiple times to explore the average and best solutions obtained by each solver, and the effects of weighting different components in the objective function are explored. The number of ambulances required per week and per station from the best solution is compared to the number known to have been utilised from data provided by QAS.

3.4.5 Contribution to the Literature

These models improve on the models in the current literature in a number of ways. A single stage model is presented that is able to minimise the number of ambulance crews needed across a planning period. Previous models minimise the number of ambulances needed for every time interval and then schedule to meet or exceed that requirement in a second stage model. Using a single stage model allows parameters such as overtime costs to be considered for the first time in an ambulance problem and allows utilisation of ambulance crews to be better understood. The real time model presents a decision tool using disjunctive constraints to schedule decisions on ambulance dispatch, relocation and meal breaks. All three models presented in this work also include the impact of hospital transfer decisions on the makespan on incidents processed by ambulances. This reduces assumptions about availability of hospital resources present in other models.

Chapter 4: Heuristics

This chapter provides a literature review and summary of a number of useful metaheuristics for scheduling problems. While not an exhaustive list, it covers some of the more relevant, popular and recent techniques which are then evaluated for suitability for the models developed in this thesis. First Come First Served (FCFS), Tabu Search (TS) and Ant Colony Optimisation (ACO) are employed to solve the models presented in this thesis, either individually or as part of a hybrid heuristic.

Basic constructive heuristics are discussed in Section 4.1; metaheuristics, including local search metaheuristics and Evolutionary Algorithms, in Section 4.2; and a discussion on hyper heuristics in Section 4.3. Table 4-1 shows the contents and relevant literature for each of these sections. Finally, Section 4.4 discusses which heuristics are selected for application in this thesis and the reasons for their selection.

Table 4-1 Summary of literature surveyed on heuristics methods

Section	**Topic**	**References**
4.1	Basic Heuristics	(French, 1982)
4.2	Metaheuristics	
4.2.1	Local Search Algorithms	
4.2.1.1	Tabu Search	(Gendreau & Potvin, 2005; Gendreau & Potvin, 2010; Glover & Laguna, 1997)
4.2.1.1.1	Applications of Tabu Search	(Brandimarte, 1993; Hurink, Jurisch, & Thole, 1994; Pitts & Ventura, 2009; Saidi-Mehrabad & Fattahi, 2007)
4.2.1.2	Variable Neighbourhood Search	(Hansen, Mladenović, Brimberg, & Pérez, 2010; Mladenović & Hansen, 1997)
4.2.1.2.1	Applications of Variable Neighbourhood Search	(Amiri, Zandieh, Yazdani, & Bagheri, 2010; Bagheri, Zandieh, Mahdavi, & Yazdani, 2010)
4.2.1.3	Simulated Annealing	(Henderson, Jacobson, & Johnson, 2003; Kirkpatrick, Gelatt, & Vecchi, 1983)
4.2.2	Evolutionary Algorithms	
4.2.2.1	Genetic Algorithms	(Reeves, 2003)

4.2.2.1.1	Applications of Genetic Algorithms	(Ak & Koc, 2012; Gao, Sun, & Gen, 2008; Jianga, Wena, Maa, Longa, & Lia, 2011; Mesghouni, Hammadi, & Borne, 2004; Nie, Gao, Li, & Li, 2012; Pezzella, Morganti, & Ciaschetti, 2008; Tay & Ho, 2008; Zakaria & Petrovic, 2012)
4.2.2.2	Particle Swarm Optimisation	(Kennedy & Eberhart, 1995; Poli, Kennedy, & Blackwell, 2007)
4.2.2.2.1	Application of Particle Swarm Optimisation	(Girish & Jawahar, 2009; Moslehi & Mahnam, 2011; Xia & Wu, 2005; Zhang, Shao, Li, & Gao, 2009)
4.2.2.3	Ant Colony Optimisation	(Dorigo & Blum, 2005)
4.2.2.3.1	Applications of Ant Colony Optimisation	(Corry & Kozan, 2004; Rajabinasab & Mansour, 2011; Xiang & Lee, 2008; Xing, Chen, Wang, Zhao, & Xiong, 2010; Yi & Kumar, 2007; Zhou, Nee, & Lee, 2009)
4.2.2.4	Harmony Search	(Yang, 2009; Zong Woo Geem, Joong Hoon Kim, & Loganathan, 2001)
4.3	Hyper Heuristics	(Burke, Kendall, & Soubeiga, 2003; Burke, McCollum, Meisels, Petrovic, & Qu, 2007; García-Villoria, Salhi, Corominas, & Pastor, 2011; Pillay & Banzhaf, 2009)

4.1 BASIC HEURISTICS

In the ambulance dispatching environment, a common heuristic approach is to dispatch the closest vehicle to the incident site (with pre-emptions allowed for life-threatening incidents). Other examples of heuristics for scheduling include Shortest Processing Time (SPT), Earliest Due Date (EDD) and First Come First Served (FCFS).

First Come First Served is a greedy heuristic that assigns the best options for each job in the order that they arrive in the system. This type of heuristic is simple to implement for dynamic as well as static problems and is able to generate feasible solutions quickly. The problem with this technique is that it doesn't consider the needs of later jobs and has been shown to be less optimal than other constructive heuristics (CH) for simple problems. For this thesis, FCFS is still of interest for hybridisation with other heuristics due to its simplicity and the way that it mimics the current process of assigning ambulances in a dynamic environment as jobs arrive.

Shortest Processing Time schedules operations in order of the total amount of processing time, with shortest processing time first. For static, single machine problems, SPT actually optimises mean flow time. It also outperforms other basic heuristics for more complicated and dynamic systems as well (French, 1982). Unfortunately, for the ambulance problem, there are flexible machines where the processing times are dependent on which machine (ambulance) is assigned to a job, due dates and a complicated objective. Under these conditions, SPT is less effective.

The Earliest Due Date heuristic is of interest for the ambulance scheduling problem because of the requirement for incidents to receive an EMS response within a given amount of time. This method schedules operations not by when they arrive, or how long they will take to complete, but in order of the due dates. It is shown to optimise static, single machine problems for maximum lateness (French, 1982). While EDD is of interest, other heuristics may be more suitable for the dynamic FFSS problem.

Pre-emption is the ability to interrupt an operation in progress in order to deal with another. This concept is important when considering a dynamic environment with jobs of different priority levels. In reality, ambulances can be diverted from their current assignment to higher priority incidents at certain times. Initially, a CH is developed for the static problem with no pre-emption allowed. Pre-emption in the dynamic and real time models is then dealt with through solving the models at successive points in time and using higher level metaheuristics to vary the order in which jobs are considered.

4.2 METAHEURISTICS

Metaheuristics are higher level heuristics for finding good solutions to optimisation problems. These computational methods are able to explore the solution space quickly and concentrate the search on areas of interest, but do not guarantee optimality. Metaheuristics are non-problem specific and can be adapted to multiple purposes, but have parameters which require tuning to improve efficiency and effectiveness of the algorithm.

4.2.1 Local Search Algorithms

This section summarises and compares Tabu Search (TS), Variable Neighbourhood Search (VNS) and Simulated Annealing (SA). All of these techniques are extensions on a classical local search. Local search algorithms search neighbourhoods around a known solution in efforts to find improving solutions. The methods discussed here require neighbourhoods to be defined appropriately and rules outlining how an incumbent solution is selected, including methods to escape from local optima in order to find global optima. A local search technique alone is not expected to be the most effective solution technique for the FFSS models proposed

in this thesis, due to the requirement to both assign and sequence incidents on flexible ambulances. Hybrid heuristics will be developed, of which a local search algorithm is only one part.

Simulated Annealing is simple to implement and the method of accepting solutions is easy to hybridise with other heuristics. However, the local search approach of the basic SA algorithm is less directed than TS or VNS. The literature shows a number of successful hybridisations with TS and other metaheuristics for solving FJSS problems. As these complex problems form the basis of the formulation of the mathematical models, TS is selected as the preferred local search method.

4.2.1.1 Tabu Search

Tabu Search is a local search metaheuristic that explores neighbouring solutions around a feasible incumbent solution where each iteration selects the best solution from the neighbourhood as the incumbent solution for the next. A tabu list contains the most recently tested solutions and prevents them from being the starting point of an iteration more than once. The algorithm returns the best result found during the search. The methodology and applications of TS are discussed in detail in Gendreau and Potvin (2005), Gendreau and Potvin (2010) and Glover and Laguna (1997).

A simple TS algorithm is presented in Figure 4-1 which uses the following parameters:

- TL the tabu list which stores forbidden moves
- TL_{max} the maximum number of moves which can be stored in the tabu list
- $N(x)$ the neighbourhood of solutions which can be reached from solution x
- $\widetilde{N}(x)$ the neighbourhood of solutions which can be reached from solution x which are not excluded by the tabu list.

Basic TS algorithm

1: Generate initial incumbent solution x
2: Store $x \rightarrow x^*$, $f(x) \rightarrow f^*(x)$, $TL \rightarrow \emptyset$
3: **while** stopping criteria = false
4: select $x = \arg\left(min_{x' \in \tilde{N}(x)} f(x')\right)$
5: **if** $f(x) \leq f^*(x)$
6: **then** Store $x \rightarrow x^*$, f(x) $\rightarrow f^*$(x)
7: **end if**
8: Add move to *TL*
9: **if** *size(TL)* > TL_{max}
10: **then** delete oldest entry in *TL*
11: **end if**
12: **end while**

Figure 4-1 Generic algorithm for Tabu Search

Stopping criteria which are useful for TS include: i) fixed number of iterations; ii) time limit; iii) number of iterations without improvement; and iv) threshold value for $f^*(x)$. The tabu list, which is essential to the purpose of TS, is short term memory containing the most recently tested moves. It is this short term memory which prevents TS cycling through solutions by preventing re-sampling of earlier solutions. This also helps TS algorithms to avoid the problem of getting stuck in local optima by accepting non-improving iterations in the event that there are no improving solutions in the neighbourhood allowed by the tabu list.

Common extensions to the basic TS algorithm presented here are aspiration criteria and probabilistic sampling of neighbourhoods. Aspiration criteria allow selection of a move forbidden by the tabu list if the solution is better than the current best solution in long term memory. This is accepted because it is clear that the solution has not yet been visited by the search. Probabilistic sampling of neighbourhoods is useful for large problems where exploring all neighbourhoods becomes extremely time consuming. Instead of exploring the entire neighbourhood, a random selection of the neighbouring solutions is explored and the best solution from these chosen as the next incumbent. This has the benefit of exploring more neighbourhoods in the same amount of time, but may miss desirable solutions.

4.2.1.1.1 Applications of TS

Tabu search is shown in the literature to be able to produce good results for Flexible Job Shops and multi-purpose machine job shop scheduling problems (Brandimarte, 1993; Dauzère-Pérès & Paulli, 1997; Hurink, et al., 1994; Pitts &

Ventura, 2009; Saidi-Mehrabad & Fattahi, 2007) where the objective is to minimise the makespan.

Brandimarte (1993) reviews the FJSS problem and introduces a hierarchical strategy to decompose the problem into the sub-problems of job to machine assignment and sequencing. Each stage is solved, each with TS.

Pitts and Ventura (2009) formulate a mixed integer linear programming model for a flexible manufacturing system and solve the assignment of tasks to resources using heuristic rules. Tabu Search is applied in a second stage to solve the scheduling of tasks. The tabu search methods in Dauzère-Pérès and Paulli (1997) and Pitts and Ventura (2009) allow each move made to either reassign the operation to a different machine or reschedule it on the same machine. Tabu Search is also used in the literature to solve problems for patient transportation, ambulance scheduling and dynamic ambulance relocation (Erdogan, et al., 2010; Gendreau, Laporte, & Semet, 2001; Kergosien, et al., 2011).

Saidi-Mehrabad and Fattahi (2007) explore the use of tabu search algorithms for the general flexible job shop scheduling problem with minimum makespan and sequence dependent set-up times. The algorithm developed by Saidi-Mehrabad and Fattahi (2007) is a two phase tabu search where the first stage sequences jobs and the second stage assigns machines. They tested their proposed algorithm against a branch and bound optimisation method for several problems of varying sizes and found that, for the tested problems of a size that the branch and bound could solve, the algorithm was able to reach near optimal solutions and the proposed algorithm was able to find solutions in reasonable time for all problems tested.

Tabu Search is simple to implement, however, neighbourhood structure and memory parameters are very important to the effectiveness of TS. In this thesis, TS has been identified for its potential applications to sequencing jobs, ability to be hybridised with other metaheuristics, and ease with which a local search approach can be modified to suit a different objective.

4.2.1.2 Variable Neighbourhood Search

Variable Neighbourhood Search is a local search metaheuristic introduced in Mladenović and Hansen (1997) for optimisation problems. A detailed description of implementation and variations of this technique is in Hansen, et al. (2010).

The basic idea is to find a local optimum in a neighbourhood and then make perturbations to move to a different neighbourhood where new solutions may be explored. Neighbourhood selection may be by descending neighbourhoods, where the best neighbour around the incumbent is selected as the new incumbent, or by random neighbourhood selection, where the new neighbourhood replaces the old neighbourhood as the incumbent if it returns a better solution. Random searches are beneficial in very large size problems to explore a large solution space faster. The algorithm for a general VNS, as described in Hansen, et al. (2010), is as shown in Figure 4-2.

Basic VNS algorithm

Generate initial incumbent solution x
while $t < t_{max}$
  $k = 1$
  while $k < k_{max}$
    select random $x' \in N_k(x)$
    find $x'' = \arg\left(min_{y \in N_k(x')} f(y)\right)$
    if $f(x'') \leq f(x)$
    then $k = 1$
      $x'' = x'$
    else $k = k+1$
    end if
  end while
end while

Figure 4-2 Generic algorithm for Variable Neighbourhood Search

One extension to VNS is a skewed search, developed to explore far-away neighbourhoods to escape from local optima over a large section of the solution space. The global best and incumbent solutions are both stored and a non-improving solution may be accepted as the new incumbent if it is a sufficient distance away from the current solution. Another extension, for Mixed Integer Linear Programming (MILP) is Variable Neighbourhood Branching, which introduces one additional constraint to define distance between solutions and hence define the neighbourhoods. This technique is able to be hybridised with other solving methods that find good solutions within neighbourhoods. Mixed Integer Nonlinear Programming (MINLP) problems have also been addressed with VNS; however, the algorithms become much more complicated in efforts to ensure that neighbourhoods are i) defined appropriately and ii) local searches return feasible solutions.

4.2.1.2.1 Applications of VNS

Variable Neighbourhood Search algorithms are applied for FJSS in Bagheri, et al. (2010) and Amiri, et al. (2010) to minimise the makespan, and in the work from (Bagheri, et al., 2010), to minimise the mean tardiness. Results show that their VNS outperforms some genetic algorithms, but not the hybrid genetic algorithm or tabu search algorithm. Variable Neighbourhood Search is not applied to solve the models in this thesis, as TS is a less complicated local search algorithm suitable for the required purpose.

4.2.1.3 Simulated Annealing

Simulated Annealing uses the concept of entropy to settle on solutions. It is introduced in Kirkpatrick, et al. (1983), with more recent developments discussed in Henderson, et al. (2003). In SA, an optimisation problem is modelled as a system initially at temperature T_0, which represents the energy in the system, with objective function value represented as entropy. The energy in the system allows it to perform hill-climbing moves. That is, non-improving solutions may be accepted according to some probability. The higher the amount of energy in the system, the greater the probability that a solution with a higher objective function value will be accepted as the incumbent solution.

Allowing non-improving solutions to be accepted allows the system to escape from local optima in order to explore new regions of the solution space. As further iterations are performed, the system cools and there is less energy to escape from minima. The theory is that by slowly lowering the amount of energy in the system, the system should settle in the state with the lowest entropy, i.e. the solution with the lowest objective function value.

The parameters required for SA are:

- T_0 Initial temperature
- α Cooling rate
- p Acceptance rate

An example of a rudimentary SA algorithm is shown in Figure 4-3. Solutions are randomly investigated around the incumbent solution. An acceptance criterion determines whether an investigated solution will be accepted or rejected as the new incumbent. Improving solutions are always accepted. Non-improving solutions may be accepted, but the probably of accepting these solutions becomes less as the system

cools. Basic stopping conditions implementable for SA are i) n iterations without improvement on solution; and ii) temperature has cooled to a final temperature T_f. The output of the algorithm is the best solution stored in memory.

Basic SA algorithm

Generate initial incumbent solution x
$T = T_0$
while stopping condition = false
  select random $x' \in N_k(x)$
  $\Delta E = f(x') - f(x)$
  $P(x', x, T) = \exp(-\Delta E/T)$
  if $P(x', x, T) \leq p$
  then $x = x'$
  end if
  $T = \alpha T$
end while

Figure 4-3 Generic algorithm for Simulated Annealing

The process of selecting new solutions to test in a basic SA algorithm is random. Where the solution space is large, as is the case in the models presented in this thesis, a random search is less desirable than a directed search that is biased toward neighbourhoods where good solutions are more likely to be found. For this reason, SA is not used in this thesis and TS or VNS are preferred as local search techniques.

4.2.2 Evolutionary Algorithms

In this section, a subset of Evolutionary Algorithms is outlined and compared. The algorithms covered are: Genetic Algorithms (GA), Particle Swarm Optimisation (PSO), Ant Colony Optimisation (ACO) and Harmony Search (HS). These metaheuristic methods invoke a memory of previous solutions to influence the search for new solutions. The better a solution, the more influence it will exert over future selections. Thus, the solution evolves from an initial random pool of solutions to converge to a good solution after the algorithm has performed a sufficient number of iterations. The techniques discussed here are all applicable to scheduling problems in some way and are able to be hybridised with a local search metaheuristics.

4.2.2.1 Genetic Algorithms

Genetic Algorithms are population based metaheuristics simulating natural selection. At each iteration, new solutions are generated from the population of previous solutions, with better solutions passing on more characteristics to the new population and random mutations introduced to explore new sections of the solution space. This is modelled through expressing solutions as chromosomes containing strings of genes. Chromosomes from two parent solutions are mixed according to a crossover condition to generate a new offspring solution, which is then subject to mutation of its genes. More successful solutions will be selected as parents more often, so that the population of solutions should eventually converge around a good solution. Further details on GA are found in Reeves (2003).

A GA should have the following parameters:

- P_0 The initial population of solutions;
- P_{iter} The population of solutions at each iteration;
- M The size of the population;
- P_c Probability of crossover occurring;
- P_m Probability of mutation occurring.

Heuristics for selecting parents based on fitness (e.g. randomly) and for crossing genes between two parents are also required. An example algorithm for GA is shown in Figure 4-4. Stopping criteria which are suitable for use with GA are: i) limited number of iterations; ii) time limit; and iii) stop when diversity of the population drops below a threshold (which requires a definition of diversity to be added to the GA).

Basic GA

```
1:   Initialise population of solutions P_0
2:   P = P_0
3:   while stopping condition = false
4:       P' = ∅
5:       while size(P') ≤ M
6:           if r ≤ P_c
7:           then  Select Parent 1 & Parent 2 from P
8:                 Offspring = Crossover(Parent 1, Parent 2)
9:           else  Parent 1 → Offspring
10:          end if
11:          if r ≤ P_m
12:          then Mutate(Offspring)
13:          end if
14:          Add Offspring to P'
15:      end while
16:      P' → P
17:  end while
```

Figure 4-4 Process for a generic Genetic Algorithm

There are difficulties with the use of GA. Firstly, the size of the population required increases as the size of the problem increases and large amounts of memory are required for large problems. Secondly, care must be taken when coding solution vectors onto chromosomes and developing crossover rules to ensure that crossover is possible and will result in feasible solutions. Thirdly, a decision must be made on how to choose the initial population. Random initialisation is possible but it may be preferable to seed the population with known good solutions, in which case another solution method must be hybridised with GA. Genetic algorithms are popular because they can be easily hybridised with other metaheuristics and because they are applicable to a wide variety of problems.

4.2.2.1.1 Applications of GA

Mesghouni, et al. (2004) show how to construct two evolutionary algorithms for flexible job shop scheduling. They evaluate the methods by running simulations of the evolutionary algorithms using a known solution as a seed for the initial population, and find that the second method provides an improved makespan in all test cases, whereas the first method found improvements only 56% of the time. This study shows that choosing a suitable representation and genetic parameters is an important step in developing an evolutionary algorithm.

Ak and Koc (2012) briefly review GAs for scheduling problems. Genetic algorithms were initially applied to scheduling problems in the 1980s and have been used to solve parallel machine scheduling and flexible job shop scheduling. For FJSP, genetic algorithms are a general search and optimisation method that create feasible solutions and mutate the order of operation and machine selection.

Pezzella, et al. (2008) show that a GA for the general FJSP with the makespan objective can be an effective solution method. Testing against benchmark problems, they demonstrate that their method can outperform other GAs with a competitive relative error from the best known solution and good convergence rate.

Zakaria and Petrovic (2012) apply GAs for reactively rescheduling flexible manufacturing systems. They define a horizon for rescheduling and generate a new solution for that horizon when new jobs arrive. The new schedule is generated with a GA and the solution is repaired within the rescheduling horizon if it is infeasible. They find that solution approaches that minimise the number of jobs which have to be reshuffled between machines provide better solutions than solution approaches based on reshuffling jobs.

Tay and Ho (2008) use GAs to evolve a set of dispatching rules for multi objective FJSPs. Several basic dispatching rules are compared. They find that basic heuristics, in particular the first in first out (FIFO) rule, are good techniques for minimising the makespan for a problem with release dates. When the objective function is to minimise tardiness or mean flow time, the evolved rules are much better solution techniques than all the basic heuristics except for earliest due date scheduling (EDD). For the objective of minimising the number of tardy jobs, the best dispatching rules were evolved dispatching rules. The results suggest that evolved composite dispatching rules, even though no rule performs well on all objective criteria, are able to outperform basic dispatching rules for multiple objective FJSPs. Dynamic FSJP with release dates have been discussed by Nie, et al. (2012) and a gene expression programming approach developed to create machine assignment rules and job dispatching rules. Their algorithm shows improved results compared to the work in Tay and Ho (2008).

Jianga, et al. (2011) also present a hybrid heuristics for solving a multiple objective FJSP formulation based on a genetic algorithm and simulated annealing that is able to converge quickly. Gao, et al. (2008) present a hybrid GA with VNS. They find that their method is able to find the same or better quality of solution than

another similar technique but took a longer computation time to solve and did not outperform the current known best solutions for the benchmark problems.

Possible applications for GA within this thesis include creating a population of existing ambulances with characteristics that can be swapped and mutated. These characteristics would include ambulance vehicle type, home ambulance station and shifts. Incidents would then be assigned to ambulances within the existing population through a constructive heuristic or hybrid local search and constructive heuristic. This approach, while possible, has two major problems. The first problem is the large amount of memory required to store a sufficiently large number of ambulances. The second problem is ensuring that each successive generation will be able to produce a feasible solution. This relates to retaining key ambulances that may be necessary feasible solutions but may not be classed as fit under poorly chosen performance measures and adequate for the complexity of shift scheduling rules. Shift schedules should be subject to crossing over and mutation to ensure that all possible options can be explored, but each newly generated shift schedule may have to change significantly in order to allow small changes without breaking the required rules.

Use of GA is opposed for this thesis due to the amount of amount of memory and the number of sub functions that require tuning in order to find good feasible solutions for the scheduling problem.

4.2.2.2 Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is a type of evolutionary algorithm, first proposed by Kennedy and Eberhart (1995) for optimisation of continuous nonlinear functions. A review and discussion of more recent applications is found in Poli, et al. (2007). The basic concept of PSO is that the solution space is populated by a set of particles which represent feasible solutions. Each particle has a position and a velocity in short term memory. Long term memory stores the best position each particle has visited. The position of each particle is updated at each iteration based on its position and velocity at the previous step, while velocity updates are based on the difference of the position of the particle to i) the best position visited by the particle at any prior step, and ii) the best position visited by any particle. This simulates a swarming effect where information from each particle is accessible by

the group and, after sufficient iterations, the solutions should converge around a good solution.

Parameters and variables in PSO are:

- Particles $\{1....I\}$ The population of solutions;
- Dimensions $\{1...D\}$ Dimensions of the search space;
- X_{id} Position of particle i in dimension d;
- V_{id} Velocity of particle i in dimension d;
- $F(X_{id})$ The fitness of position X_{id}
- P_{id} Best known position visited by particle i in dimension d;
- P^g_d Best position visited across all particles in dimension d;
- ω Inertial Weight (affects increase or decrease in velocity at each step);
- c_1 Cognition learning factor (affects the update of velocity w.r.t. the best known solution);
- c_2 Social learning factor (affects the update of velocity w.r.t. other particles);
- r_1, r_2 Random numbers uniformly distributed on [0,1].

The particles are initialised as random feasible solutions. At each iteration, the position of each particle updates according to its previous position and velocity. The velocity is then updated based on information about the swarm of particles as a whole. By updating velocity based on group properties, with particles at good positions exerting more pull, the particles begin to converge around a good solution. These steps are shown in Figure 4-5, based on the process described in (Poli, et al., 2007).

Particle Swarm Optimisation has also been successfully hybridised in the literature. However, PSO, similar to GA, requires a larger population (of particles) for larger problems. This scaling of the size of the swarm presents potential issues for the size of the problems that are to be solved in this thesis. For this reason, other metaheuristics are preferred.

Basic PSO algorithm

1: Initialise population of random feasible solutions X_{id}
2: **while** stopping condition = false
3: **for each** particle *i* and dimension *d*
4: $X'_{id} = V_{id} + X_{id}$ [Update position]
5: $V'_{id} = \omega V_{id} + c_1 r_1 (P_{id} - X_{id}) + c_2 r_2 (P_d^g - X_{id})$
6: **if** $F(X_{id}) < F(P_{id})$
7: **then** $X_{id} \rightarrow P_{id}$
8: **end if**
9: **if** $F(X_{id}) < F(P_d^g)$
10: **then** $X_{id} \rightarrow P_d^g$
11: **end if**
12: **end for**
13: **end while**

Figure 4-5 Generic algorithm for Particle Swarm Optimisation

4.2.2.2.1 Applications of PSO

Girish and Jawahar (2009) apply a particle swarm optimisation (PSO) algorithm to the FJSP. They compare their results with best known results against a set of problems in the literature and against a constraint programming formulation. They find that their PSO algorithm achieves solutions closer to the best known solution than the constraint programming formulation (for the makespan objective), and suggest further improvements can be made by creating a hybrid heuristic using PSO and local search technique.

Particle Swarm Optimisation may also be hybridised. Xia and Wu (2005) present a hierarchical solution approach and use hybrid heuristics to solve a multi-objective flexible job shop scheduling problem. Their hybrid heuristic of simulated annealing (SA) and particle swarm optimisation (PSO) was used to solve the FJSP, with assignment and scheduling considered as interdependent rather than separately solvable. Their results show that their hybrid heuristic equals or outperforms the compared techniques for problems tested with full flexibility of resources. Moslehi and Mahnam (2011) develop a hybrid PSO and local search algorithm and produce competitive results but are not able to outperform existing solutions. Zhang, et al. (2009) use a hybrid heuristic for the multiple objective FJSP, incorporating particle swarm optimisation and the tabu search metaheuristic, which performs well at minimising the makespan.

4.2.2.3 Ant Colony Optimisation

Ant Colony Optimisation (ACO) is an agent-based solution approach which mimicks the way that ants place and interpret pheromones to communicate which paths are worth following (Dorigo & Blum, 2005). For a scheduling problem, assignment and sequencing decisions are arcs on a disjunctive graph. These 'trails' may be traversed by 'ants' and evaluated on their contribution to good outcomes. Exploring a large number of solutions reinforces good decisions by placing additional pheromone each time an arc is visited. The amount of pheromone placed on an arc depends on the objective value of the solution to which the arc contributes. Pheromone also evaporates over time so that poorer decisions are visited less frequently.

Ant Colony Optimisation is useful for strategic solutions for ambulance scheduling and shift scheduling and can be hybridised with other heuristics to make the approach more suitable for smaller planning horizons. The concept, adapted for FFSS and the scale of the objective function values, has the following characteristics:

- τ_0 A limiting parameter fixing the maximum amount of pheromone that may be present on any disjunctive arc;
- U A contribution parameter that influences the scale of pheromone applied relative to the performance of the solutions;
- α A parameter affecting the amount of pheromone remaining from old trails and the amount of pheromone applied from new trails;
- R The probability that the arc with the most amount of pheromone will be selected ($0 \leq R \leq 1$);
- η A heuristic for evaluation of each arc independently of pheromone. Suitable functions for this heuristic include response times or costs such as overtime;
- β A parameter for the emphasis placed on the value arising from heuristic function η;
- δ Power scaling of C_q and C_{best} for determining contribution from each arc traversed.

The steps for determining arc selection based on pheromone includes a calculation of the probability of selecting an arc where job *j* is assigned to machine *m* from the set of all possible machine assignments *M* (Equation (4.1)). A second equation that determines whether the arc selected is the best known arc or random arc (Equation (4.2)). This second equation reinforces good decisions but allows unexplored decisions to be considered and mitigates the risk of falling into local optima.

$$Prob(j,m) = \frac{Trail(j,m) \times \eta(j,m)^{\beta}}{\sum_{n \in M} Trail(j,n) \times \eta(j,n)^{\beta}} \qquad \forall\, j \in J, m \in M \tag{4.1}$$

$$j = \begin{cases} \arg\left(\max_{m \in M}\left(Trail(j,m) \times \eta(j,m)^{\beta}\right)\right), & r \leq R \\ S, \ otherwise \end{cases} \qquad \begin{matrix} \forall\, j \in J, \\ m \in M \end{matrix} \tag{4.2}$$

The scale of pheromone to be applied to visited arcs is calculated according to Equation (4.3). This considers the fitness of all the solutions to which the arc contributed in the latest iteration and then adds an additional term to consider the contribution in the overall best solution.

$$\Delta Trail(j,m) = \sum_{q \in K} \frac{U}{C_q} + c_{jm} \frac{U}{C_{best}} \qquad \forall\, j \in J, m \in M \tag{4.3}$$

The amount of pheromone is updated with each iteration according to Equation (4.4). The total amount of pheromone is the amount of pheromone remaining after evaporation plus the new pheromone laid down during the latest iteration, or the maximum amount of pheromone allowed, whichever is less.

$$Trail'(j,m) = \min(\tau_0, (1-\alpha) Trail(j,m) + \alpha \Delta Trail(j,m)) \qquad \begin{matrix} \forall\, j \in J, \\ m \in M \end{matrix} \tag{4.4}$$

4.2.2.3.1 Applications of ACO

One example of the utilisation of ACO is Yi and Kumar (2007), where ACO is used to solve a mixed integer network flow model for disaster relief. This type of problem manages the flow of material between nodes, and Yi and Kumar (2007) specifically consider transportation of wounded people from demand nodes to hospital nodes. Response times and severity of wounds are taken into account. Disaster relief operations differ from daily ambulance operations in that a transportation of multiple people in a single trip may be more convenient and several returns to a single node may be required. They find that ACO heuristics are useful

for speedy solutions, which is vital because a planner in disaster situation needs obtain and update solutions for routing and assignment quickly. An optimality gap exists but is small enough to be acceptable for the gains in runtime.

Xiang and Lee (2008) apply ACO for dynamic manufacturing with multiple resources and products and flexible machines. This is an agent based approach to solving a dynamic scheduling model. The algorithm performs well for some, but not all, objectives in the case study. For example, the solutions from the ACO algorithm are superior to FIFO on tardiness objectives but inferior on makespan. Further tuning of the algorithm is suggested by the authors.

Zhou, et al. (2009) use ACO for dynamic job shop scheduling and show that ACO is applicable to the problem. However, the ACO algorithm proposed does not outperform appropriately chosen dispatching rules. Rajabinasab and Mansour (2011) also use an agent based approach and apply it to the dynamic FJSP. A comparison of results against dispatching rules using discrete event simulation suggests that the agent based algorithm proposed by Rajabinasab and Mansour (2011) is more robust than other approaches in environments with high machine utilisation.

Ant Colony Optimisation for FJSS is approached in Xing, et al. (2010) with a multiple phase approach. The first stage arbitrarily selects an operation and then assigns a machine using the probability determined by accumulated knowledge. The second stage sequences the sets of operations on each machine to create a feasible schedule. At each time t when a machine is idle, the operation from the allowable set with the highest probability of being sequenced at that point is selected. A schedule-improving heuristic, involving crossing over components of good solutions, is performed after each iteration and pheromone is deposited prior to the next iteration. The best solutions found from the ACO based heuristic presented in Xing, et al. (2010) offer slight improvements for the cases tested when compared against other heuristics published in the literature. However, the average solutions and CPU runtime are not provided. The paper shows that ACO is applicable to FFSS and can produce good results but there is room for further improvements and testing of this approach.

The ACO algorithm considered for application in this thesis uses ideas presented in Corry and Kozan (2004). They tackle a machine layout problem for flexible machines where i) machines must be positioned, and ii) an order in which

machines are selected for position must be determined. Order for all machines is determined first, and then a second stage positions machines, because the positioning of a machine restricts subsequent feasible positions. The paper demonstrates that the best and average solutions obtained by the ACO algorithm are an improvement on previous solutions to benchmark problems, including more consistent machine layouts over multiple horizons. Unfortunately, the ACO algorithm requires a greater computational effort, suggesting that the approach is more suitable for planning horizons where machine relocations do not happen frequently (e.g. daily horizons may be suitable where hourly horizons are not).

4.2.2.4 Harmony Search

The Harmony Search (HS) metaheuristic draws inspiration from methods used by musicians improvising in order to create harmonies (Yang, 2009; Zong Woo Geem, et al., 2001). Decision variables may be considered as 'instruments' with allowable integer values as the 'notes' which may be played. The musical notes played on each instrument in order to create good harmonies overall are decided at each step, by: remembering and returning to notes that worked well; shifting the note played to a neighbouring note; or, randomly selecting a new note to play.

The components for developing a HS based algorithm are:

- *Harmony Memory* (HM) which stores the *'x'* best solutions that have been tried;
- *Harmony Memory Consideration Rate (HMCR)* specifying how frequently notes are selected from memory as opposed to randomly;
- *Pitch Adjustment Rate* (PAR) which determines how often a note selected from memory will be adjusted to a neighbouring note;
- *Bandwidth* (BW) that defines the extent of the neighbourhood for pitch adjustment.

The use of bandwidth in the literature is usually introduced as a linear function for adjusting pitch, for example $x_{new} = x_{old} + BW \times \varepsilon$, (Yang 2009). Pitch adjustment rate and bandwidth have implications for the convergence of solutions and the area of the solution space that is explored. Low PAR and narrow bandwidth focus the search near known solutions, while higher PAR and broader bandwidth will explore the solution space faster but act more like a random search.

The steps of HS follow the sequence of initialisation, improvisation and evaluation. Improvisation selects a note from memory if a random number is less than the HMCR. In this case, if a second random number is less than the PAR, the new note will be adjusted to a random note within the bandwidth of the note from memory. If no note is selected from memory, a random note will be generated. New solutions are added into HM if there is space available in the memory. Once HM has the maximum number of allowable solutions, new solutions are only added if they outperform an existing solution. In this event, the poorest solution is removed from memory to make space for the new solution. This process is shown in Figure 4-6.

Basic HS algorithm

Generate initial feasible solutions to populate HM
while stopping condition = false
  if $r \leq HMCR$
  then $x^{'} \in HM$
    if $r \leq PAR$
    then $x_a^{'} \in BW(x_a^{'})$
    end if
  else Generate random $x^{'}$
  end if
  if $f(x^{'}) \leq \max_{y \in HM}(f(y))$
    $x' \rightarrow HM$
    remove $x = \arg(max_{y \in HM}(f(y)))$ from HM
  end if
end while

Figure 4-6 Generic algorithm for Harmony Search

4.2.2.4.1 Applications of HS

Harmony Search has the potential to be hybridised with a local search technique to create a hybrid heuristic to solve the models in this thesis. Incidents, representing instruments, are able to be assigned to ambulances which represent musical notes, while another heuristic varies the sequence in which incidents are selected for assignment. This method requires a pool of ambulances to be initialised (with associated ambulance stations and shifts) so that the bandwidth may be defined as all neighbouring ambulances suitable to respond an incident. This requires a large amount of memory. Alternatively, incidents might be considered as the notes placed in position for consideration in a hybrid heuristic where another heuristic assigns ambulances. Harmony Search is not used in this thesis but it is a possible extension for developing additional hybrid or hyper heuristics.

4.3 HYPER HEURISTICS

Metaheuristics search within a search space of problem solutions. Hyper heuristics search within a search space of heuristics to choose the best heuristic to use to solve a particular optimisation problem. They are introduced into the literature because heuristic techniques have strengths and weaknesses in different scenarios (Burke, et al., 2003; García-Villoria, et al., 2011). Several heuristic components (or a set of simple heuristics) may be combined and adapted within a hyper heuristic which will generate a search technique and learn from each iteration and exploit that knowledge in the next. The hyper heuristic solution method is more adaptable than a tuned heuristic and may be applicable in more than one problem domain.

4.3.1 Applications of Hyper Heuristics

Hyper heuristics can broadly be divided into constructive and improvement heuristics that either build a solution or improve an initial feasible solution. Both types of hyper heuristic are investigated in García-Villoria, et al. (2011), with sets of simple heuristics and metaheuristics forming the basic components of the hyper heuristic. García-Villoria, et al. (2011) use hyper heuristics for the response time variability problem, an NP-hard scheduling problem that models the situation where several resources are required for a task and the objective is to minimise the variability of the time between arrival points of resources. They show that hyper heuristics can be competitive for NP-hard problems compared to solutions from basic heuristics, and note that including more sophisticated metaheuristics into hyper heuristics is a path that merits further investigation.

Hyper heuristics have been used for both job shop scheduling and personnel scheduling. In the scheduling environment, they have been used for exam scheduling problems (Pillay & Banzhaf, 2009) and educational time tabling problems (Burke, et al., 2003; Burke, et al., 2007). For the timetabling problems in the two studies of Burke et al., the hyper heuristic approach works well on medium and small problems but is still outperformed by a tailored ACO algorithm for larger problems. However, the hyper heuristic was able to obtain feasible solutions in more scenarios.

The process of developing an appropriate hyper heuristic for the ambulance scheduling problem first requires an exploration into which basic heuristics and metaheuristics provide good solutions. If the performance of metaheuristics or hybrid heuristics is dependent on a scenario then a hyper heuristic may improve

performance. However, due to the additional computational power required for a hyper heuristic, this technique may not be applicable for an online solution, and a hybrid heuristic is the preferred solution approach for this thesis.

4.4 SELECTION OF HEURISTICS

This section summarises the suitability of the discussed heuristics for application in this thesis and proposes a methodology for hybridisation of heuristics.

Any heuristic algorithm applied to the ambulance and ambulance crew scheduling problems must be able to guarantee a feasible solution. Heuristic algorithms which alter existing solutions by moving a job from one ambulance to another, or moving an ambulance to a different shift or ambulance station, may end up exploring a lot of infeasible solutions due to constraints on response time windows. For example, incident *i* assigned to ambulance *a* might feature in a feasible solution but swapping incident *i* to different ambulance *b* may result in an infeasible schedule if other incidents, scheduled on ambulance *b*, already occupy the response time window for incident *i* and cannot be rescheduled or reassigned to another ambulance. Identifying feasible swaps has the potential to require a large amount of computational power and may be slow to explore the solution space as a consequence. For this reason, heuristics which can quickly construct a feasible schedule are favoured over heuristics which take a feasible schedule and make alterations.

Constructive heuristics can be used to assign resources to incidents in an order defined by rules such as FCFS, SPT and EDD. First Come First Served is chosen for all of the CHs developed in this thesis because this is a realistic approach for dealing with dynamic data. Applications of metaheuristics for constructing feasible solutions are also considered. Ant Colony Optimisation is identified as a suitable solution-building metaheuristic because it is able store preferential information on decision arcs to influence the selection of resources when constructing solutions. It has the benefit of exploring more of the decision arcs than a basic CH but requires a larger amount of memory and time. This heuristic is described and applied in Chapter 7.

Metaheuristics are also considered for hybridisation with a constructive heuristic. Hybrid heuristics may consist of a combination of any two heuristics. For the ambulance problem, the hybrid heuristics explored contain a constructive

heuristic and a metaheuristic. A constructive heuristic is chosen for its simplicity, speed and the guarantee that it will return a feasible solution. It is hybridised with a metaheuristic to allow a greater region of the solution space to be explored. The methodology used to combine these is shown in Figure 4-7. This shows an outer heuristic varying the order in which jobs are considered by the inner heuristic, which is the CH that builds each solution. The CH is required to be fast as it is called for every solution explored by the metaheuristic. The inner and outer heuristics developed for each individual model are described in more detail in later chapters.

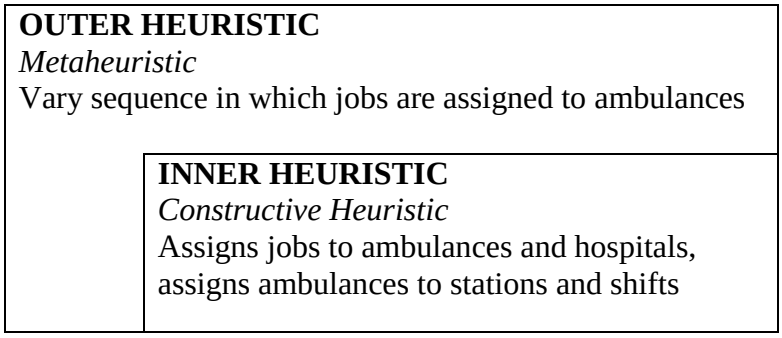


Figure 4-7 Structure of the proposed hybrid heuristic

The metaheuristics tested for the outer heuristic are ACO and TS. Both of these metaheuristics have been successfully applied to FJSS problems in the literature. Selecting one local search algorithm (TS) and one evolutionary algorithm (ACO) allows the benefits and disadvantages of each approach to be explored. Tabu Search is simple to implement if a suitable neighbourhood definition can be established. It is less complicated than VNS and more directed than SA. As such, TS is the preferred local search algorithm for hybridisation. Ant Colony Optimisation is selected as the preferred evolutionary algorithm for a hybrid heuristic partially because it can be also be used to construct solutions without hybridisation. This makes it ideal for comparing the effectiveness of a metaheuristic against a hybrid heuristic. Additionally, investigation of the literature suggests that ACO may have lower memory requirements than GA and PSO. These metaheuristics require a large population of solutions to influence the selection of future solutions. ACO places markers on decision arcs which accumulate as more solutions are found, without the requirement to maintain many solutions in memory at the same time. Harmony Search is less developed than the other metaheuristics. It has been relegated to future

work if hybrid heuristics are shown to be successful through testing a better known metaheuristic (i.e. ACO).

Multiple heuristics and hybrid heuristics are tested. The purpose of testing several heuristics is to begin with a simple solution approach which is improved each time a new concept is introduced.

A hyper heuristic requires additional computational power and suitable selection of heuristic algorithms from which it can draw. Suitably tuned hybrid heuristics are preferred as they are expected to be faster to solve.

Chapter 5: Case Study

This chapter discusses the data available on EMS in Queensland and the relevance of EMS in Brisbane to the research proposal. Historical data on ambulance services is analysed in order to extract parameters that describe the demand for EMS. The reason for, and process of, generating a new set of data from these parameters is explained, and this new data set compared with the real data.

The remainder of this chapter is set out as follows: Section 5.1 introduces the available data sets; Section 5.2 discusses the information which may be derived from the data sets and is of use for testing the optimisation models; Section 5.3 provides a methodology for extracting parameters to define a new, anonymous, data set; Section 5.4 generates the new data set; and Section 5.5 validates the new data set against real data. Section 5.6 outlines the shift scheduling rules used with the case study in the models developed in Chapters 6, 7 and 8.

5.1 ENVIRONMENT

This thesis focuses on EMS in metropolitan and semi-urban areas, and the case study concentrates on the Brisbane metropolitan area in Queensland, Australia. There are approximately 1000 incidents each day across the Brisbane region, of which around 700 are urgent or emergency incidents, and the average cost per incident is $520 (Queensland Ambulance Service, 2014). There is an annual upward trend in the total number of incidents (Queensland Ambulance Service, 2014; Queensland Treasury, 2007).

Following a restructure during 2012/2013, the entire state of Queensland encompasses 15 contiguous regions and 298 ambulance stations containing EMS, Patient Transport Service and Intensive Care Paramedic units. The Brisbane region is split into the Metro North (21 ambulance stations) and Metro South (14 Ambulance stations) regions.

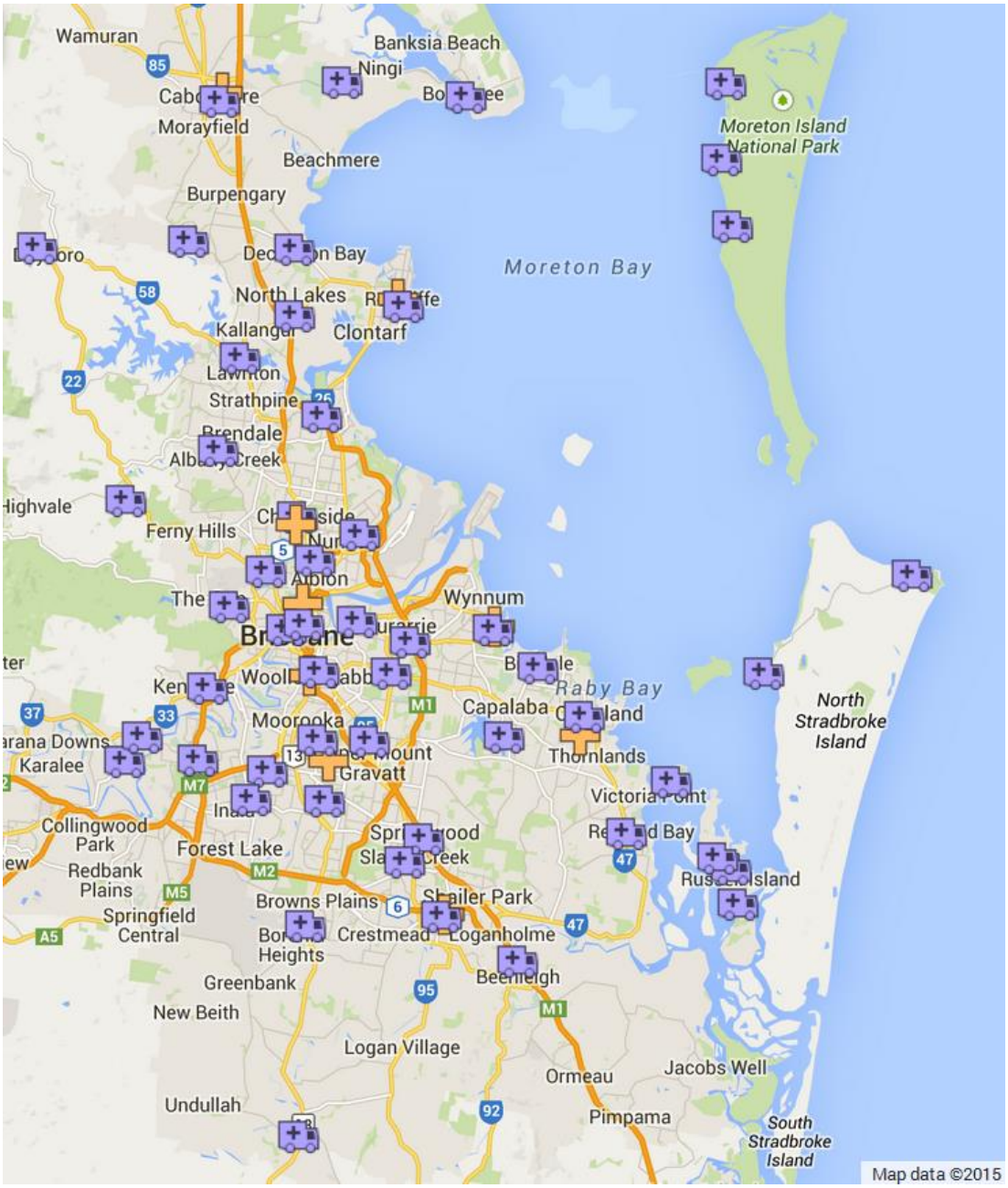


Figure 5-1 Ambulance stations (represented by blue vehicles) and public hospital locations (denoted by orange crosses) across the Brisbane metropolitan area

The ambulance stations in the Brisbane region are plotted in Google Maps and shown in Figure 5-1. Locations for public hospitals with emergency departments within the Brisbane region were extracted from Queensland Health (2013) and plotted on the same map. Only 30 stations of the 35 in the Brisbane region were considered of interest for the model as the other five are too remote. Thirty ambulance stations is still a large set of data for an optimisation model case study. In order to test each model in reasonable time, a smaller set containing five ambulance stations within a central area is selected. This is further discussed in Section 5.3.

5.1.1 Available Data

Two sets of data were analysed for this thesis. The first set is Queensland Ambulance Workforce Modelling Data from 2003/2004 to 2006/2007. This provided simple information on the number of incidents of different priority types handled by different ambulance stations across regions and the number of ambulances scheduled at each station across each hour of the week. The second set of data requested from and provided by QAS, is incident data for the 2011/2012 Financial Year. This contains more detailed and up to date information on incidents handled by QAS, including various timing points for each call. Publicly available reports on ambulance performance are also examined.

5.1.1.1 Workforce Modelling Data

Queensland Ambulance Workforce Modelling Data contains information on ambulances within the Brisbane region in the 2003/2004, 2004/2005, 2005/2006 and 2006/2007 working years. This includes the hourly resources available at each station for each year. Files also include the number of incidents handled by each ambulance station at each hour of the seven day working week and the average 'time-to-clear' for each station. Incidents were separated into three triage categories: Emergency (Code 1); Urgent (Code 2); and Non-Urgent (Code 3+4). Additional information provided with these files contained the percentage of combined priority 1 and 2 calls that were met in < 10 minutes for each hour of the day during July 2007. Exact location of each call was not given in the available data.

5.1.1.2 Incident Data for the QAS 2011/2012 Financial Year

The next set of data was provided from QAS in May 2013. This is a large data set containing information on over 340,000 unique incidents arising in the Brisbane

metro and surrounding regions for the entire 2011/2012 Financial Year. This contains the following information:

- Incident identifier: a unique code given to each incident that arrives in the system;
- Incident date: the date and time at which an incident was logged;
- Priority code: the assessed priority of the incident;
- Timing points: the clock time at which events were recorded as occurring;
- Latitude and longitude: the location of the incident (covering an area of southern QLD including Brisbane).
- Cancel reason: a reason given for cancelling an event (or NULL).

In further detail, the timing points contained information on the date and time that:

- incidents were received;
- an ambulance was dispatched to an incident and/or an ambulance was recorded as on the case of an incident;
- an ambulance arrived on the scene of an incident;
- an ambulance departed the scene of an incident;
- the ambulance arrived at the next destination (i.e. hospital);
- the time an ambulance was again available and/or the incident was cleared.

This information represents arrival rates, spatial distribution, priority type distribution, processing times and the number of incidents requiring further transit to a hospital.

5.1.1.3 Ambulance activity

Publicly available information on performance indicators is available online (Queensland Ambulance Service, 2014) and in reports *(Queensland Department of Community Safety, 2012; Queensland Treasury, 2007)*. These publications explore trends and record performance measures such as number of incidents and response times. This provides additional background information on ambulance services and may prove useful for validating the models.

It is this data that provides evidence on the increasing number of incidents each year. Table 5-1, showing daily activity for QAS, indicates that the number of incidents has increased by slightly less than 5% per year for the last 3 years across Queensland. The increase is for all types of incidents, although emergency and urgent incidents make up the majority of ambulance incidents. Table 5-2 shows daily activity across the Brisbane metropolitan region is increasing faster than the state wide average, and incidents within the metropolitan region account for nearly half of all the incidents in Queensland. Furthermore, the rising trend in the number of incidents each year extends back for more than a decade (Figure 5-2).

Table 5-1 Daily ambulance activity across QLD

Year	2007/08	2008/09	2009/10	2010/11	2011/12	2012/13	2013/14
Total Number of Incidents	1978	2038	2041	2180	2277	2384	2456
Emergency and Urgent Incidents	1377	1376	1415	1533	-	1737	1806
Non-Emergency Incidents	601	662	626	647	-	641	649
Increase from previous year	-	3.03%	0.15%	6.81%	4.45%	4.70%	3.02%

Table 5-2 Daily ambulance activity across the Brisbane metropolitan region

Year	2010/11	2011/12	2012/13	2013/14
Number of Incidents	768	786	987	1048
Emergency and Urgent Incidents	-	-	706	734
Non-Emergency Incidents	-	-	282	313
Increase from previous year	-	2.34%	N/A (region redefined)	6.18%

Figure 5-2 also shows that ambulance responses (the number of ambulances actually dispatched to incidents) exceed the number of incidents and are also increasing. Responses surpass incident numbers because of reassignment of ambulances and because some incidents require multiple ambulance vehicles. Emergency incidents have the highest response-to-incident ratio and there is evidence that this ratio is increasing slightly over time for emergency and urgent incidents *(Queensland Treasury, 2007).* Increases in urgent and emergency incidents are significant challenges to ambulance services.

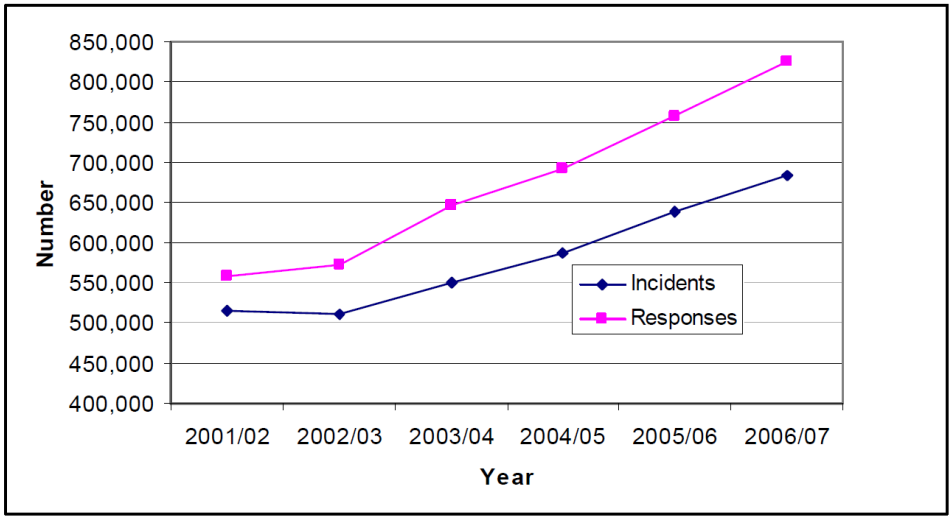


Figure 5-2 Annual ambulance incidents and responses across QLD. Source: *(Queensland Treasury, 2007)*

5.1.2 Shift Scheduling Rules

The following shift scheduling rules for ambulance crews are taken from ambulance workplace agreements and are relevant to this case study (Queensland Industrial Relations Commission, 2012). These are to be incorporated progressively into the models presented in the next chapters in this thesis.

- **Time off between shifts.** A minimum period of 8 hours rest must occur between the last time worked and the beginning of the next shift.
- **Forward rostering is preferred.** In order to combat fatigue, scheduled shifts should be arranged such that a shift must start at the same time of day, or later, than the previous shift except where a day off has occurred.
- **Limit on consecutive night shifts.** No more than two nights shifts may be worked sequentially.
- **Minimum rostered days off.** A minimum of two full days off must be granted each week such that a 48 hour period from midnight to midnight occurs with no scheduled hours.
- **Meal breaks.** Meal breaks must be assigned each shift such that a thirty minute break is scheduled between 4 and 6 hours into the shift. An additional rest period of 20 minutes should also be scheduled at any point during each shift.

Figure 5-3 displays the time window permissible for meal breaks. Figure 5-4 contains an example of feasible schedules meeting the shift scheduling rules described.

5.2 ANALYSIS OF ACTUAL DATA

Using the information sources presented above, the distribution of incidents is analysed. The models developed in this thesis require information about incidents in order to be tested. The models with deterministic data require the following pieces of information to be available at the initialisation stage:

- the number of incidents;
- the time at which each incident is available in the system;
- the priority (i.e. triage category) of each incident;
- the hospital transfer preferences associated with each incident;
- estimates of processing times on the scene of an incident;
- estimates for time spent ramping, admitting patients to a hospital and cleaning ambulances after an incident;
- incident locations;
- ambulance station locations;
- hospital locations;
- estimates of travelling times between locations; and
- beginning and ending times for ambulance shifts.

The real time model requires the same information but, where relevant, to be updated in response to the current state of the system. Some of these parameters may be interdependent or time dependent. In this section, parameters describing incident distributions are investigated and extracted to later be used for generating new incident data sets.

Duplication of incident identifiers, more commonly for emergency and urgent incidents, exists in the original data. One cause of duplication for these incidents is that multiple ambulances may be dispatched to life threatening cases in order to make sure one will arrive quickly. Duplication can also occur if multiple ambulances are required at the scene or if a previously assigned ambulance was reallocated and a different dispatch required to be made. Duplicate data is retained

for analysis of demand patterns but, for simplicity, removed from the data used to extract processing times.

Duplicate incidents are not counted when extracting hospital transfer rates and processing times. Only a record with hospital transfer data will be selected as the main record for the duplicated incident. Duplicates with shorter dispatch to clear times are excluded from use in extracting processing times under the assumption that a shorter time represented ambulances which were dispatched to an incident but did not remain with the incident until it was clear. Nonsensical values for timing points, for example instances where the clear time was before the dispatching time, are also removed or fixed where found.

Removing duplications and nonsensical data reduced the data set by over 24%. There remains a risk that incorrect data has not been removed or that correct data from duplicate incidents passed over. Assumptions for selection of which data record is to count as the 'original' also risks inflated processing times. To moderate this risk, care must be taken in selecting appropriate distributions to generate processing times for the data used in the model

Figure 5-3 Example time window for meal breaks

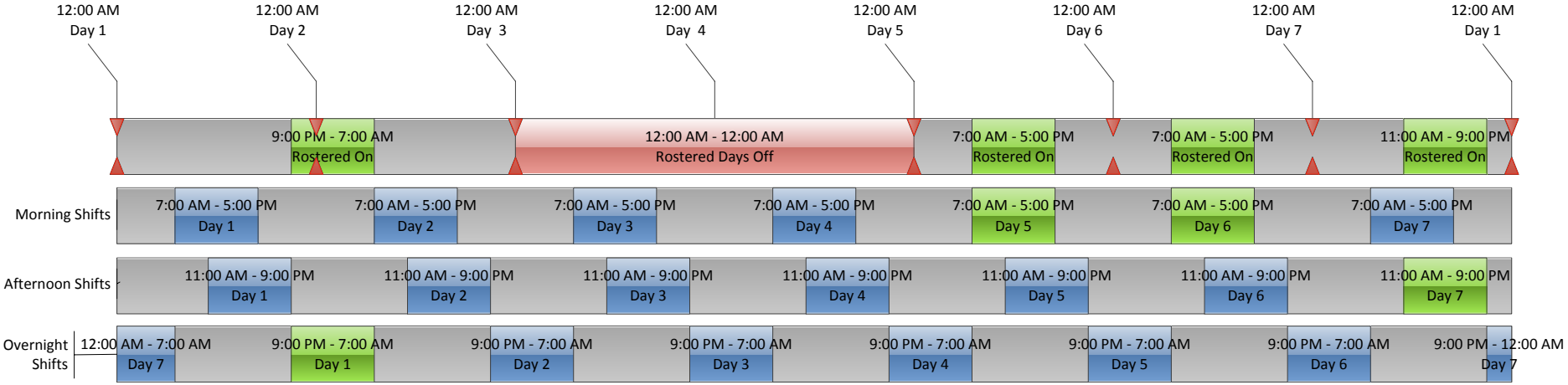


Figure 5-4 Example of feasible weekly shift schedule obeying all rules

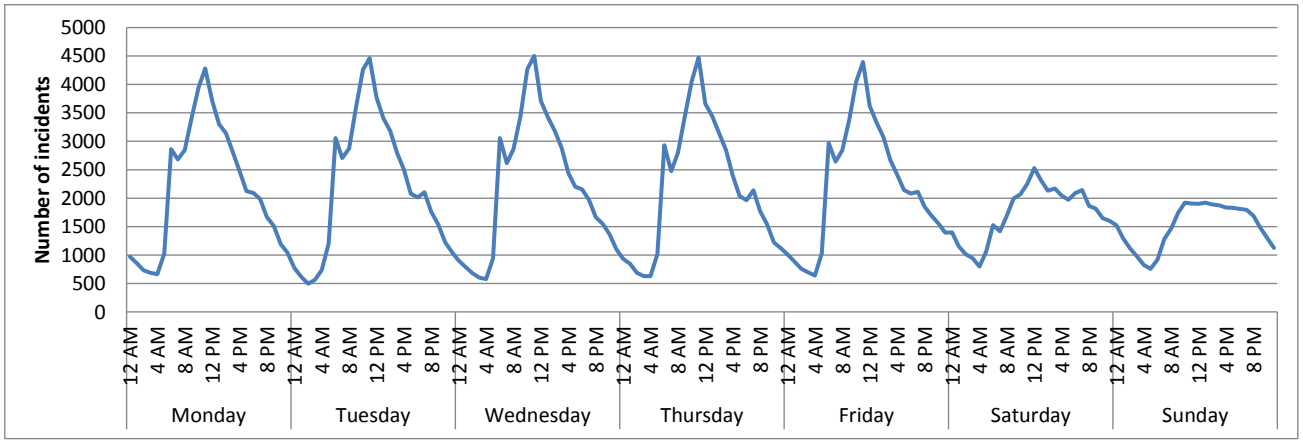


Figure 5-5 Ambulance incidents per hour across Brisbane for 2011/2012

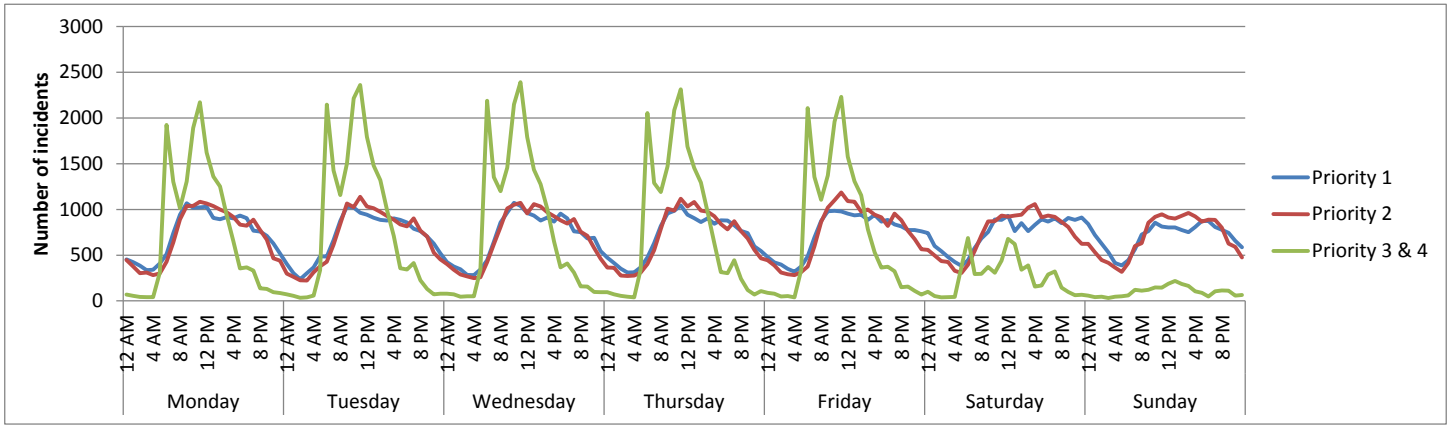


Figure 5-6 All ambulance incidents across Brisbane for 2011/2011, split into priority types for each hour of the week

5.2.1 Seasonality

The demand for ambulances is found to have strong 24 hour and weekly seasonality, as shown in Figure 5-5. Cycles involve a daily pattern on weekdays (Monday to Friday) with morning peaks and smaller afternoon peaks coinciding with the beginning and ending of workdays. The lowest demand for ambulances occurs a few hours after midnight. Weekend (Saturday and Sunday) demand has a flatter profile than weekday demand with much lower peaks and less dramatic overnight lulls. Seasonality is important as it impacts on the number of ambulances that will need to be scheduled at each hour of the day, each day of the week and, as such, plays an important role in determining shift starting times.

5.2.2 Priority Type

The 2011/12 QAS incident data is analysed as shown in Figure 5-6. Emergency and urgent cases (Priority 1 and 2) outnumber non-urgent cases and are the driving force for EMS daily and weekly seasonality. Priority 3 and 4 (non-urgent) include patient transport services provided by ambulance services, commonly to and/or from hospitals, which may require paramedics to travel with a patient. All of these incidents are considered for this thesis.

Incident analysis from 2011/12 data confirms that, while all priority types follow daily and weekly seasonality, emergency/urgent incidents have different demand profiles to non-urgent incidents. Non-urgent incidents have multiple daily peaks on weekdays (morning and midday) and much lower demand during evenings and weekends. This matches with expectations of transport to and from appointments at hospitals, or transfer between hospitals, which would be scheduled during regular working hours.

Data used to verify the models should reflect the differences in arrival rate between emergency/urgent incidents and non-urgent incidents.

5.2.3 Demand Distributions

Data from 2011/12 is investigated to highlight areas where demand is concentrated within Brisbane both spatially and temporally. Figure 5-7 is a contour plot, which is created in Matlab from available data, to survey the spatial demand profile across the Brisbane region. Darker colours represent a higher density of calls on a logarithmic scale. Hospitals are plotted with yellow stars and ambulance

stations with magenta triangles. To create this plot, longitude and latitude for incidents throughout the entire 2011/12 year were used to place each incident onto a square spatial grid. The plot shows demand is centralised around the inner city, where population density is high, and smaller hotspots correlate with the locations of hospitals. This information is useful, as it highlights areas within the metropolitan region where coverage rates need to be highest. This affects parameters which need to be used for the real time model.

Figure 5-8 is a similar contour plot focused on the busy inner north area, bounded by latitude [153, 153.1] and longitude [-27.475, -27.350], within the metropolitan region. This density in this region is explored further in Figures 5-9, 5-10 and 5-11, which show the density for each of the three different priority codes. Code 1 (emergency) and Code 2 (urgent) incidents are found to occur more evenly through the area, with greater frequency in areas with high population movement, whereas Code 3 (non-emergency) incidents are more focussed at the hotspots located near to hospitals. Contour plots for each hour of the week indicate that the location of demand hotspots does not change with time; however, the importance of these hotspots is highest at times of peak demand.

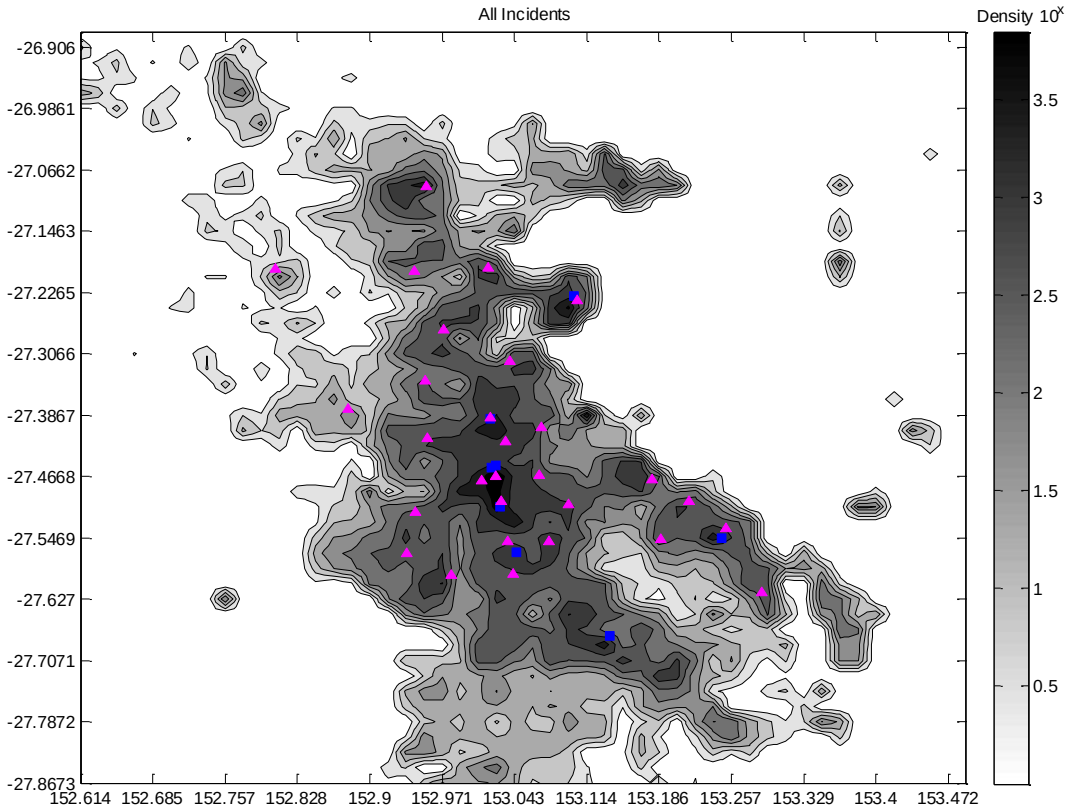


Figure 5-7 Density of demand across Brisbane for 2011/12

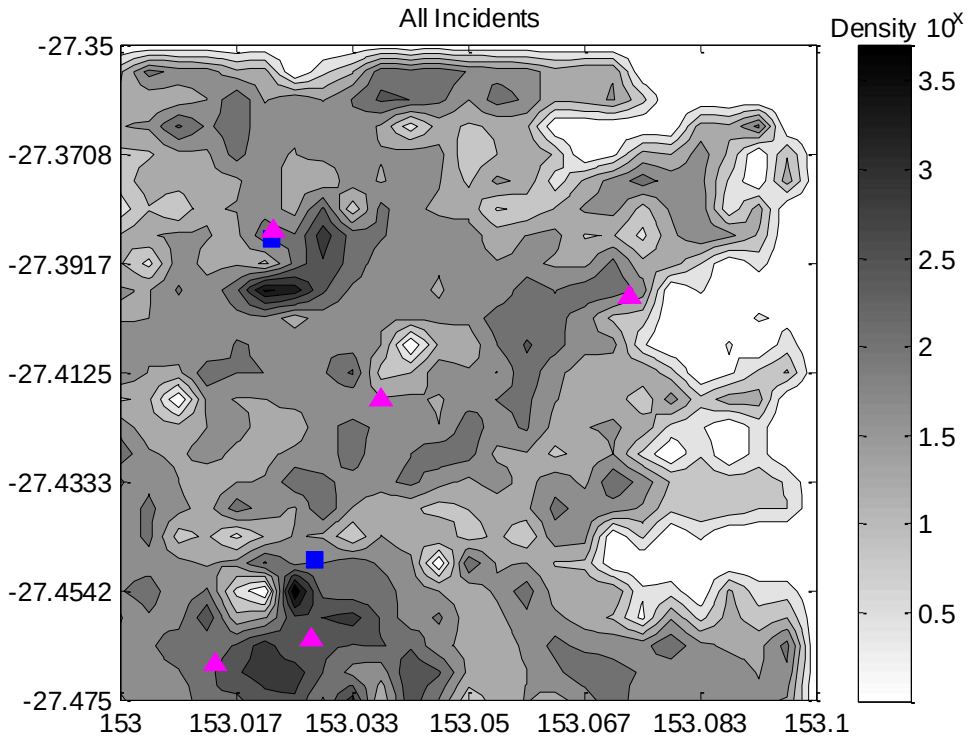


Figure 5-8 Demand density in the busy inner northern region of Brisbane, 2011/12

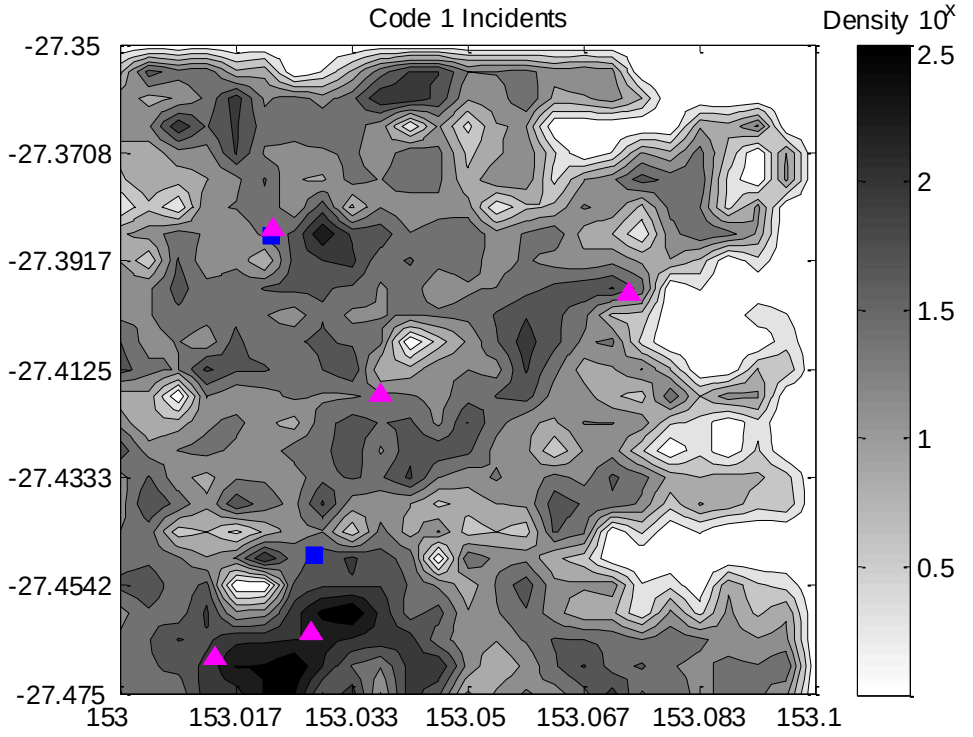


Figure 5-9 Code 1 demand density in the busy inner northern region of Brisbane, 2011/12

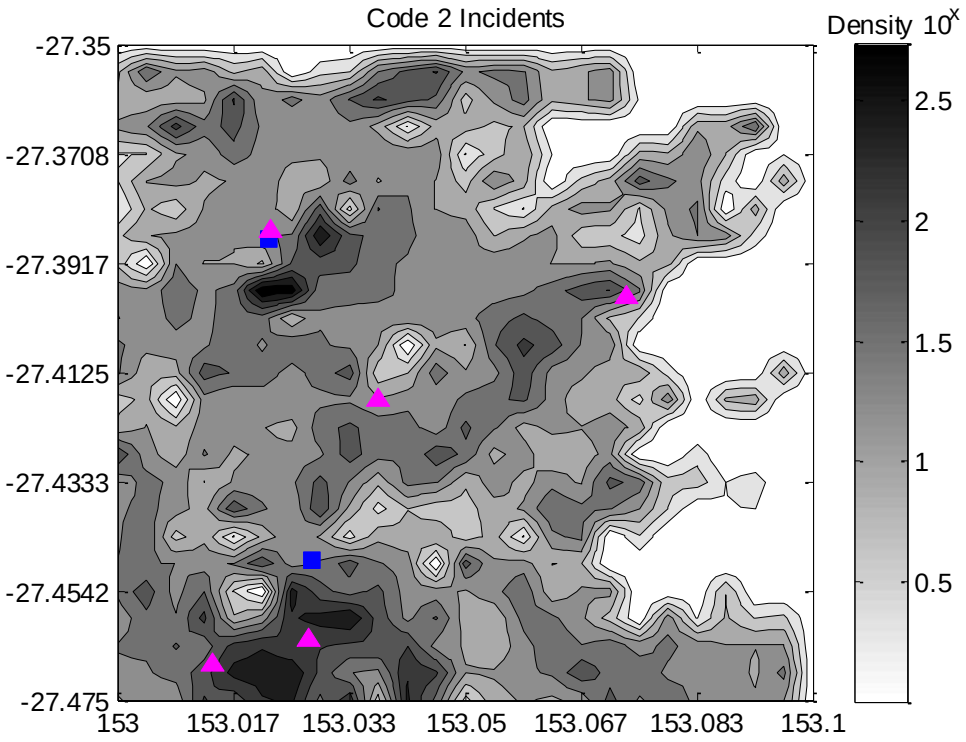


Figure 5-10 Code 2 demand density in the busy inner northern region of Brisbane, 2011/12

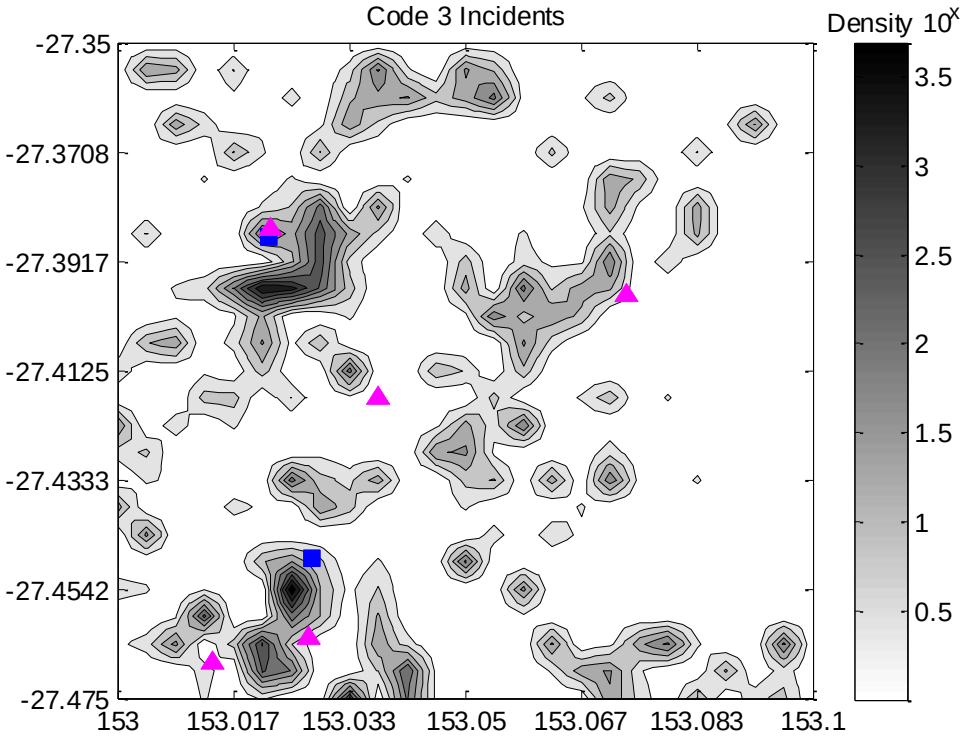


Figure 5-11 Code 3 demand density in the busy inner northern region of Brisbane, 2011/12.

5.2.4 Response Times

Tables 5-3 and 5-4 show response times from public performance data for emergency calls across QLD and the Brisbane metropolitan area respectively. Response times have been steady across QLD for the past seven years, with 50% of emergency incidents met in around 8.2 mins or better. The response time for the best 90% of incidents is smaller within the metropolitan area than the across the entire state, although this is not observed for the best 50% of incidents

Table 5-3 Emergency response times across QLD (in minutes)

Year	2007/08	2008/09	2009/10	2010/11	2011/12	2012/13	2013/14
50^{th} Percentile	8.3	8.4	8.1	8.2	8.3	8.2	8.2
90^{th} Percentile	16.7	17.2	16.4	16.7	17.0	16.5	16.3

Table 5-4 Emergency response times across the Brisbane metropolitan area (in minutes)

Year	2011/12	2012/13	2013/14
50^{th} Percentile	8.6	8.5	8.4
90^{th} Percentile	16.7	15.7	15.4

Response times, based on the time from dispatch to the first response arriving on scene, are extracted from the 2011/12 QAS incident data. The response times extracted are shown in Table 5-5, which contains the 10^{th}, 50^{th} and 90^{th} percentiles for response times for each of the three priority types.

Table 5-5 Percentile response times from the 2011/12 incident data

Priority	Response time (mins)		
	10^{th} Percentile	*50^{th} Percentile*	*90^{th} Percentile*
ALL	2.3	10.6	46.3
Code 1	2.1	7.1	13.9
Code 2	2.1	11.1	24.5
Code 1 and 2	2.1	8.5	20.2
Code 3	3.7	30.3	91.8

As expected, Code 1 incidents have the fastest response times, followed by Code 2 and then Code 3. Compared with the response time percentiles from QAS public information, results from the incident data provided underestimates the

response time. This is possibly due to measuring the response time from the time of the first dispatch and not from the time at which an incident becomes available. However, response time from dispatch to arrival is useful for verification of the models presented later in this thesis. Another explanation of the difference between the 2011/12 data and the public performance data is the inclusion of cancelled incidents in the 2011/12 data. Cancelled incidents have been retained in the data used to create the new data for the case study. These incidents still require an ambulance response to be sent until the cancellation has been logged. For the models presented in the next section, cancelled incidents are treated as incidents with appropriately short processing times.

The variations in response time per hour of the day are highlighted in Table 5-5. Emergency and urgent incidents maintain steady response times throughout the day, while non-emergency incidents have a fluctuating response time, with higher response times beginning just before and ending just after normal business hours. This may have a relationship with times at which hospital appointments begin and finish. In conjunction with the spatial distribution of non-emergency incidents near to health care facilities, it is concluded that a large proportion of the non-emergency incidents in the data available are actually patient transportation for appointments which may be scheduled in advance. Scheduled transportation and dynamic demand are not separated in the available data. For this reason, and because the models developed should schedule all demand, transportation incidents are treated the same as incidents arising dynamically in this thesis.

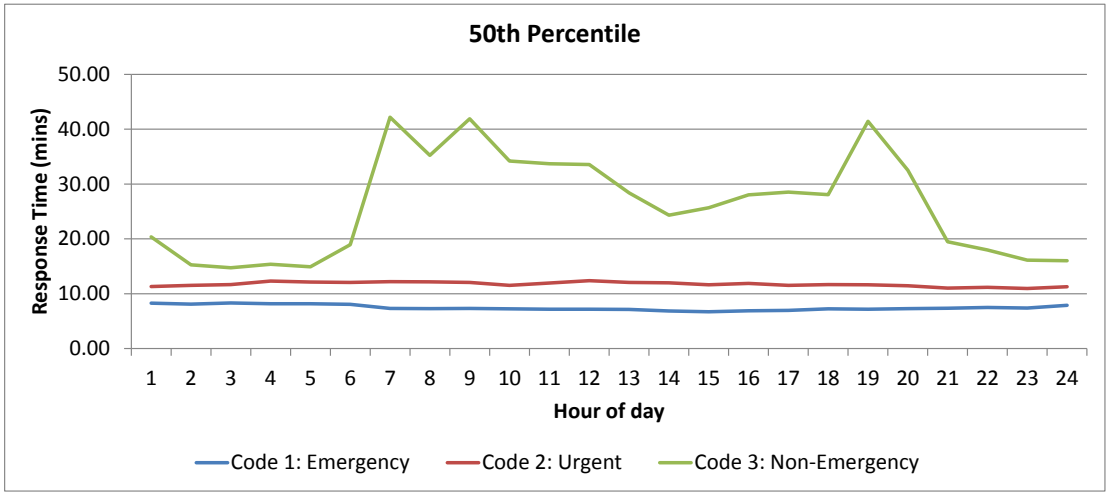


Figure 5-12 Daily 50th Percentile response time for incidents of each priority type

5.2.5 Dispatch to Clear

Time to clear is an important performance measure for ambulance services. Reducing dispatch to clear times would increase the number of incidents that each ambulance is available to respond to, particularly during peak demand periods. Performance reports display this information for urgent and emergency cases (seen in Table 5-6), indicating that the average time an ambulance is busy with an incident is between 70 and 80 minutes. For comparison, dispatch to clear times determined from the 2011/12 incident data, for all priority types, are in Table 5-7. Firstly, it is noted that non-emergency incidents have a higher average dispatch to clear time than emergency and urgent incidents. Secondly, the values from the 2011/12 data are higher than from the performance reports. This may be due to a combination of improved performance from 2011/12 to 2012/13 and discrepancies in the extraction of data from the incident file.

Table 5-6 Average time from dispatch to clear for emergency and urgent incidents

Year	Jul 2012 – March 2013	Jul 2012 – March 2013
Statewide (QLD)	72.9 mins	69.0 mins
Metropolitan area (Brisbane)	79.2 mins	72.9 mins

Table 5-7 Dispatch to clear times extracted from QAS 2011/12 Incident data

Priority Codes	Mean (mins)	Median (mins)
All	95.0	89.2
Emergency and urgent	91.4	88.1
Emergency only	93.7	90.4
Urgent only	89.2	85.5
Non-emergency	103.6	93.1

Figure 5-13 shows average dispatch to clear time for the 2011/12 data as a function of time. This figure indicates that dispatch to clear times have a daily seasonality pattern which is apparent for all priority codes. Peaks occur in the middle of each day, during times when demand is also highest. This makes sense because increased travel delays from traffic conditions during the day and longer ramping times at hospitals during peak demand periods would be expected to cause longer dispatch to clear times for ambulances.

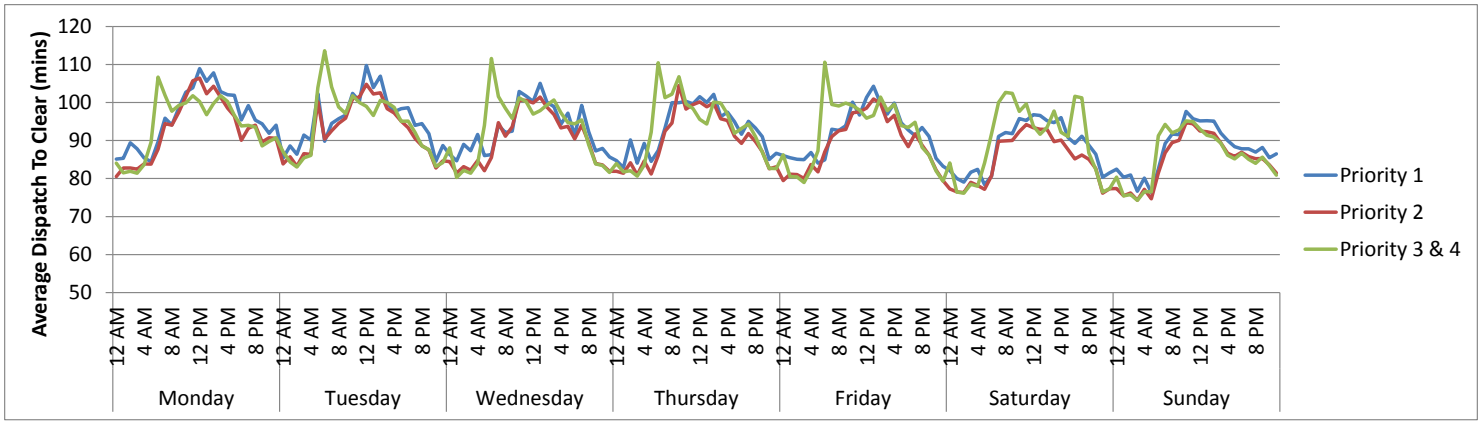


Figure 5-13 Daily dispatch to clear time for each priority types for 2011/12 incident data

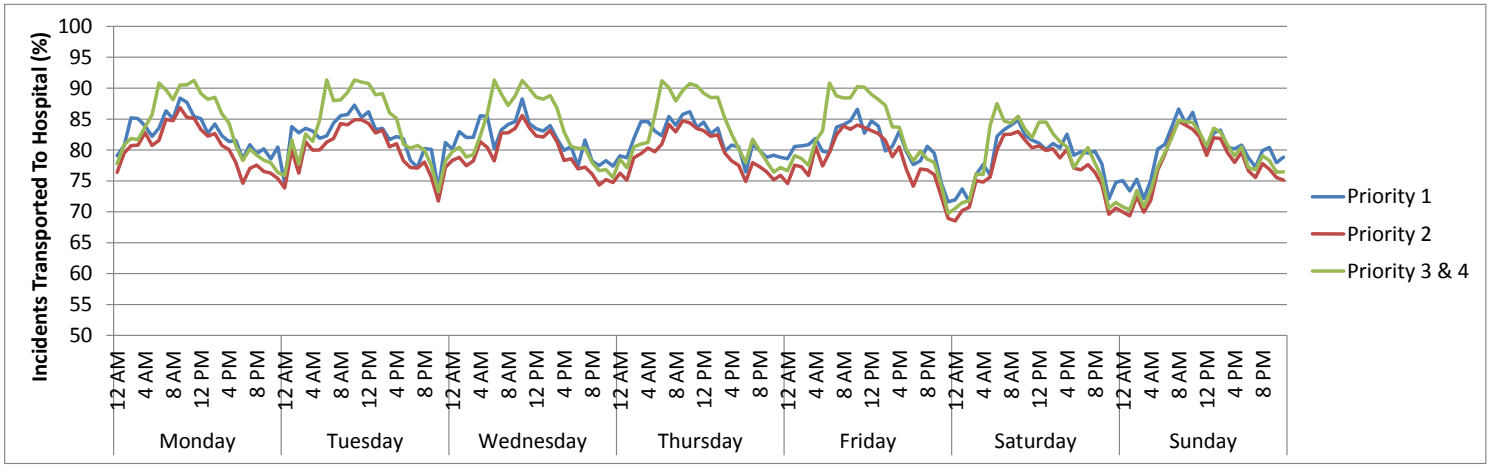


Figure 5-14 Percentage of ambulance responses resulting in further transportation

5.2.6 Time on Scene

Time on scene is defined as the amount of time between when an ambulance is recorded as arriving at the scene of an incident and when they depart. This time may be spent assessing the situation, treating and/stabilising a patient, transferring a patient into the back of an ambulance if necessary or other tasks at the scene. QAS incident data records "d_on_scene" and "d_depart_scene" against incident identifiers. This information is used to test the distribution of time spent at the scene for all incidents and priority types. Results are shown in Table 5-8. Emergency calls require the greatest amount of time on scene. The difference between the median and mean signifies skewness from a small number of outliers where large amounts of time are required to be spent on scene for an incident, most noticeably for non-emergency incidents.

Table 5-8 Time spent on scene as extracted from QAS 2011/12 Incident data

Priority Codes	Mean (mins)	Median (mins)
All	19.52	15.38
Emergency and urgent	21.72	17.97
Emergency only	22.37	19.67
Urgent only	19.75	16.15

5.2.7 Hospital transfers

Not all patients are transported to hospital after receiving a response from an ambulance. The majority, however, are transported to a further location. Table 5-9 shows the percentages of incidents which are transferred from the scene by an ambulance from values in QAS publications (Queensland Ambulance Service, 2012). Between 80% and 90% of ambulance responses require transportation, with a slightly higher percentage in the metropolitan area than the state wide average. Analysis of hospital transfer rates by priority type and time is possible using the 2011/12 incident data (Table 5-10 and Figure 5-14).

Hospital transfer rates show some dependence on incident priority type. Non-emergency incidents have the highest number of hospital transfers because these include patient transportation services (e.g. transportation of patients to hospitals for appointments or transfer of a patient from one hospital to another). Emergency incidents have the next highest rates for hospital transfers, followed by urgent incidents. Daily patterns also exist, with peaks in the middle of the day, highest for non-emergency incidents, when most schedulable transportation would occur. The average transportation rate from this data, covering mostly

the metropolitan region with some surrounding areas, is 84.4% (and 81.3% for emergency incidents), suggesting a slight increase in the hospital transfer rate over the last three years.

Table 5-9 Patients transported by ambulance to another location from public performance information

Year	**2012/13**	**2013/14**
All incidents		
Statewide (QLD)	84.2%	86.3%
Metropolitan area (Brisbane)	86.8%	87.4%
*Emergency and urgent incident*s		
Statewide (QLD)	-	86.4%
Metropolitan area (Brisbane)	-	86.6%

Table 5-10 Percentage of hospital transfers by priority type from 2011/12 incident data

Priority	All incidents	Code 1 Incidents	Code 2 Incidents	Code 3 Incidents
Percentage of incidents transferred to hospital	81.45%	71.04%	77.08%	94.48%

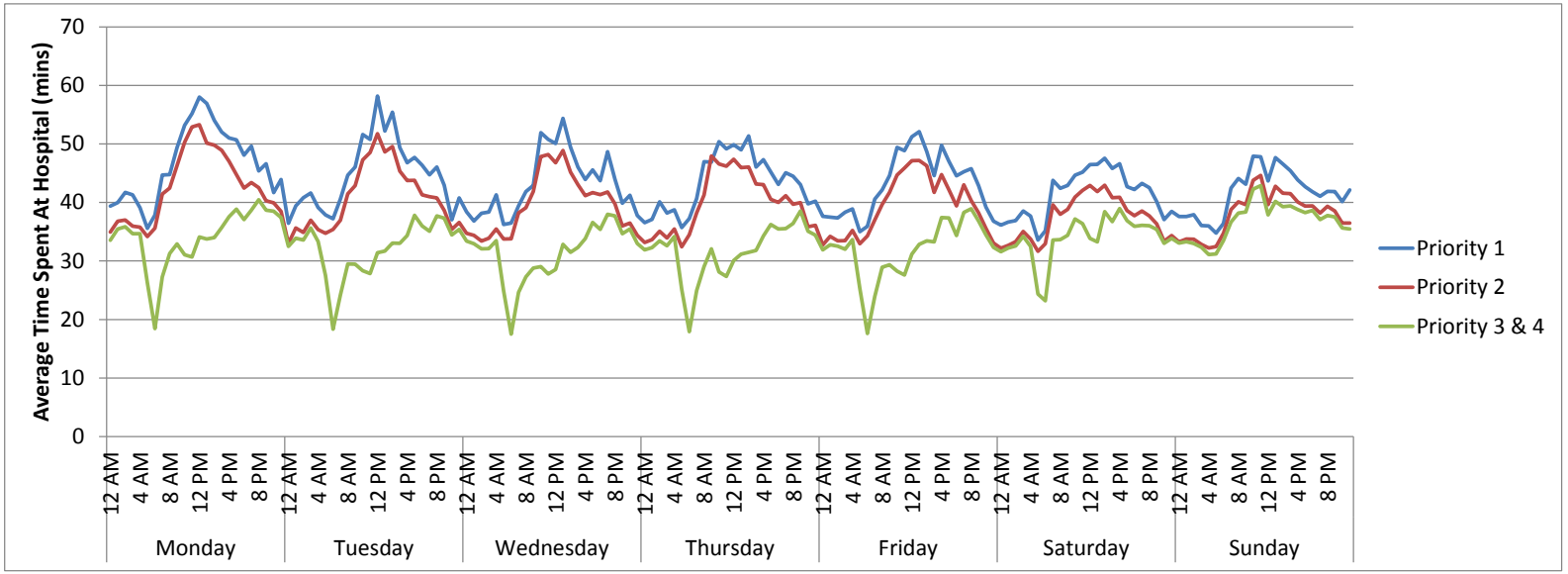


Figure 5-15 Average time between arriving at a hospital and being cleared

5.2.8 Time at Hospital

For this section, time spent at a hospital is inclusive of ramping time, time to admit the patient and time to clean an ambulance if necessary. This time is found through the different between "d_at_destination" and "d_clear" from QAS 2011/12 data. Analysis for time at hospital follows the same process described in the previous section for time on scene, with additional incidents removed where no transfer to hospital occurred. Table 5-11 presents the average time spent between arriving at hospital and an ambulance becoming clear for each priority type. Figure 5-15 extends this to each hour of the week.

Table 5-11 Arrival at hospital to clear times extracted from QAS 2011/12 Incident data

Priority Codes	Mean (mins)	Median (mins)	80th percentile (mins)	<30 mins
All	39.5	34.78	59.5	43.7%
Emergency and urgent	**53.0**	**46.1**	**68.6**	**14.4%**
Emergency only	56.2	50.0	72.2	14.3%
Urgent only	50.0	42.1	63.9	25.3%
Non-emergency	12.9	8.7	58.3	91.1%

It was expected that high priority incidents would be admitted into hospitals faster than other incidents. Contrary to this expectation, the results showed that non-emergency incidents were admitted faster. As Code 3 non-emergency incidents include patient transfer between facilities and patient transportation to appointments, not all non-emergency incidents have to pass through the Emergency Department. Congestion in the ED is the major cause of delays in admitting patients arriving by ambulance into hospitals. By circumventing the ED, non-emergency incidents return a faster admission time.

It was also found that admissions to hospital occur faster at the beginning of a working day and become slower throughout the day. Code 1 and Code 2 incidents, categorised by ambulance triage as emergency and urgent incidents, pass through the emergency department, which is a prime source of ramping delays when at or near capacity. The longest average waiting times correspond with periods of peak demand. Average times are also skewed by a small number of incidents taking extreme amounts of time (several hours) to be admitted into a hospital, an event that is known to occur.

This thesis considers ambulances process and performance measures; however, these are affected by the performance of hospital emergency departments. The performance measure at the interface between emergency departments and ambulances is the percentage of patients transferred off-stretcher in 30 minutes. Current values for this performance measure are around 70–80% with targets of 80% (Queensland Health, 2013). The QAS data for emergency and urgent incidents from 2011/12 does not reflect this information. This is partially because of policies implemented by healthcare services in QLD over the last few year to reduce ramping time, or redefine the way ramping time is measured, making a comparison between 2011/12 and 2014 data inaccurate. Data from September 2012 has two major hospitals in the area, Prince Charles Hospital (PCH) and Royal Brisbane and Women's Hospital (RBWH), with off stretcher percentages of 56% and 67% respectively, confirming that large improvements have been made in this sector (see Table 5-12)

Table 5-12 Emergency Department performance for two selected hospitals in the Brisbane metropolitan region

Hospital	Off stretcher within 30 mins	
	September 2012	August 2014
PCH	56 %	79 %
RBWH	67 %	80 %

Off stretcher times from 2012 still vastly outperform the data extracted from the QAS incident data. While there is still an effect of improvements being made from 2011 to 2012, there is also the possibility that some of the outliers in the data, causing the extreme time to clear from arriving at a hospital, are due to errors in the data file itself and/or the process used to extract data.

5.3 GENERATING NEW DATA

A new set of data is necessary to test the model. Ethical reasons preclude the use of real location data that can link people to requests for medical aid. Therefore, parameters are extracted from the real data and used to generate a new set of anonymous incidents. A new data set is generated to be an anonymous data set for a small case study within the Brisbane metropolitan area covering a time period of two weeks. Five ambulance stations and two public hospitals were selected from the busy

inner north region of Brisbane. This area was selected as it covers an area where incident density is extremely high. Ambulance stations included are Spring Hill and Chermside, very busy stations; Roma Street and Northgate, moderately busy ambulance stations; and Kedron Park, a location especially set up to provide fast first responses to high priority incidents. The five stations chosen for the small case study were selected from 2007 data on ambulance station locations. Since then, some ambulance station locations across QLD have changed. An ambulance station at Keparra has been decommissioned and replaced by two new stations at Mitchelton and Ashgrove, in close proximity to the case study. Additional stations have been added at Pinjarra Hills and Archerfield and the operational regions changed so that some ambulance stations falling outside of the Brisbane region in 2007 are now part of the Metro South or Metro North regions. Future work may look at extending the case study to include more ambulance stations, including the newer stations.

The two major public hospitals in the case study within the inner north region of Brisbane are the RBWH and PCH. Other hospitals exist in this area, such as the children's hospital and smaller private hospitals. For the purposes of simplification, these are not included as they are in close proximity to the major two public hospitals with emergency departments and specialist units in the case study. Hospitals outside of the area of interest are already excluded. The capacity of emergency departments at hospitals to admit ambulance patients is considered, for this thesis, as an external parameter unaffected by ambulance decisions. While this is not true in reality, because ambulances transporting a large number of patients to a single hospital can stretch its capacity and increase ramping times, this effect is out of scope for this project and is left for future work.

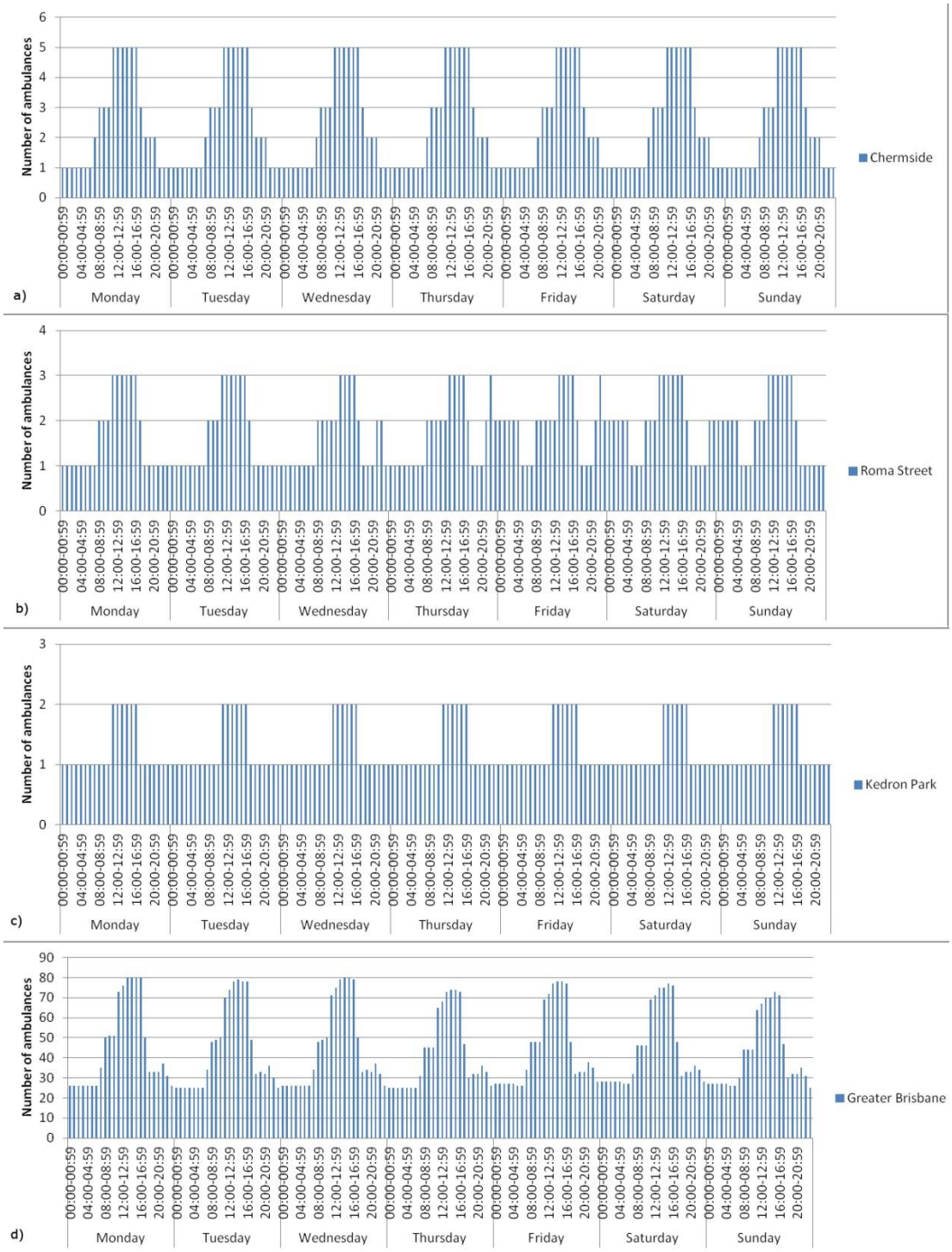


Figure 5-16 The number of ambulances working each hour for various stations across Brisbane

5.3.1 Shift Patterns

The models developed in this thesis use standard shift patterns to assign ambulances. In order to solve these effectively, a simplified but realistic shift pattern is needed. This pattern is extracted from the number of ambulances working in the Brisbane region and then compared against a small number of stations.

Workforce utilisation data from 2006/2007 is investigated to look at current shift patterns. Figure 5-16 shows the number of ambulances working per hour for a) a busy station (Chermside); b) an average station in the Brisbane area (Roma Street); and c) a small, specialist station (Kedron Park). Figure 5-16 d) shows the number of ambulances working each hour for the entire greater Brisbane region. It can be seen that each of these plots reflects the seasonality of the demand with daily peaks and lulls.

To create a small sample of feasible shifts in the case study, with which to build a full shift schedule, the following process is used. It is assumed that ambulance shifts have a uniform duration of 10 hours. In theory, there are 24 possible times that shifts may begin each day (i.e. on the hour, every hour) and therefore 168 possible shift beginnings each week. A simple IP optimisation model is developed to match workforce data to the possible shifts. This model is outlined below.

Parameters

H Set of hours in a week, $h \in [1, 168]$

S Set of shifts in a week, $s \in [1, 168]$

A_h Number of ambulances working at hour h

$$B_{hs} = \begin{cases} 1, \text{if shift } s \text{ covers hour } h \\ 0, \text{otherwise} \end{cases}$$

Decision Variable

C_s Integer number of ambulances assigned to shift s

Objective Function

The objective minimises the difference in the number of ambulances working each hour on the shifts that are pre-set in the model and the actual number of working ambulances recorded.

Minimise

$$\sum_{h \in H} \left| \sum_{s \in S} (B_{hs} * C_s) - A_h \right|$$

Constraints

The number of ambulances assigned to shifts must be zero or positive and integer

$C_s \geq 0$ and integer

The optimal solution to this problem yields an objective of 44 ambulance hours. This is a difference of less than 1% of the total number of ambulance hours for the week. Less than half of the 168 possible shifts across the week were selected. Weekly shifts were then collapsed into daily shifts and sorted by the number of ambulances allocated to shifts beginning at each hour of the day. Table 5-13 shows the shifts used and how many ambulances were placed onto each. The shaded rows on the table are the shifts required to make sure each hour of the day is covered by at least one shift.

Shifts are selected from the table in descending order until a full 24 hour period is covered. This requires only the top three shifts on the list: a morning shift from 7am to 5pm; an afternoon shift from 11am to 9pm; and a night shift from 9pm to 7am. Nearly 70% of all ambulances are allocated to one of these three shifts. This shift pattern is shown in Figure 5-17. The fourth and fifth shifts on the list are the first and third shifts, respectively, delayed by one hour. Adding in these two offset shifts boosts the percentage of ambulances falling within the simplified shift pattern to nearly 90%. This pattern is shown in Figure 5-18.

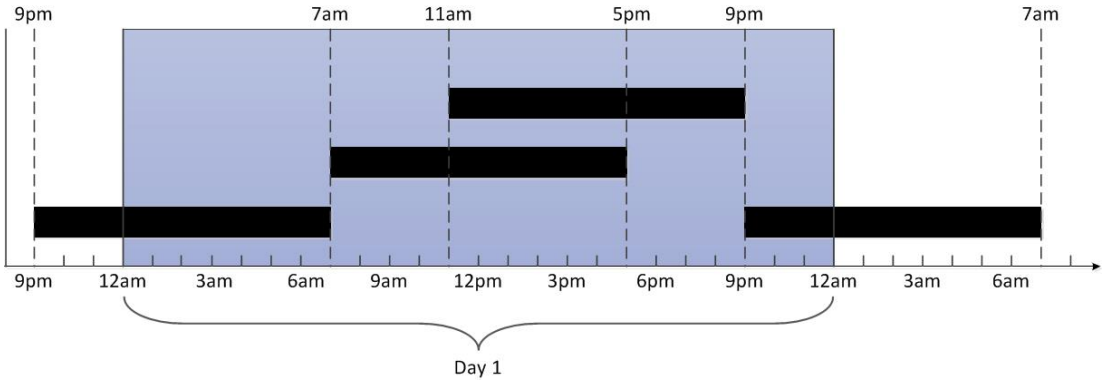


Figure 5-17 The most simplified effective shift pattern covering a full 24 hour cycle

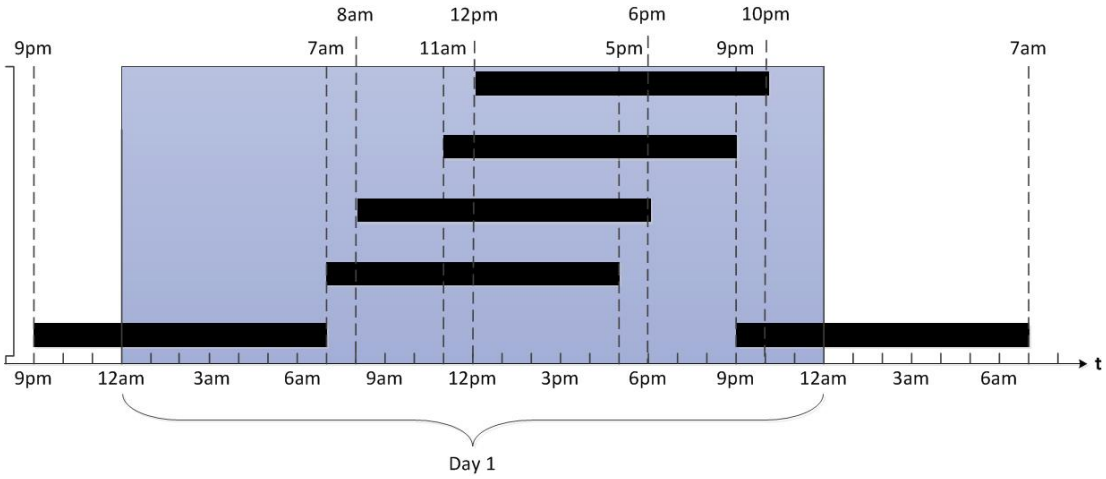


Figure 5-18 Example of an effective shift pattern with two additional shifts more than the most simplified pattern

To determine which shift is most appropriate, the simplified patterns are compared for the small selection of individual stations from the case study: Chermside, Roma Street and Kedron Park (see Table 5-14). The shift pattern for Kedron Park can be met exactly through the use of the three shift cycle, and Chermside is met exactly with the five shift cycle. Roma Street and the Greater Brisbane Region utilise more flexible shifts in reality, and cannot be matched with shifts of fixed 10 hour duration.

The models tested in the case study use the simplified pattern with three daily shifts of 10 hour duration. This is chosen because it is the most efficient pattern for the 24 hour cycle

and can match the workforce data exactly for at least one ambulance station. The five shift cycle improves the match to workforce data but increases the decision variables in the mathematical models. This amplifies the difficulty of finding solutions. Exploring the effects of additional shifts is left for further work.

Table 5-13 Shifts identified from 2006/2007 workforce data with shaded rows representing the minimum requirements to cover a 24 hour period without gaps

Shifts		*Ambulances allocated*			*Cumulative percentage of ambulances allocated*		
Start	*Finish*	*All*	*Weekdays*	*Weekends*	*All*	*Weekdays*	*Weekends*
07:00	17:00	205	150	55	28.08	28.41	27.23
21:00	07:00	158	114	44	49.73	50.00	49.01
11:00	21:00	134	96	38	68.08	68.18	67.82
08:00	18:00	111	78	33	83.29	82.95	84.16
12:00	22:00	37	27	10	88.36	88.07	89.11
13:00	23:00	36	27	9	93.29	93.18	93.56
22:00	08:00	10	7	3	94.66	94.51	95.05
17:00	03:00	7	4	3	95.62	95.27	96.53
03:00	13:00	6	4	2	96.44	96.02	97.52
19:00	05:00	6	3	3	97.26	96.59	99.01
10:00	20:00	4	4	0	97.81	97.35	99.01
14:00	00:00	4	3	1	98.36	97.92	99.50
05:00	15:00	3	3	0	98.77	98.48	99.50
06:00	16:00	2	2	0	99.04	98.86	99.50
20:00	06:00	2	2	0	99.32	99.24	99.50
15:00	01:00	2	1	1	99.59	99.43	100.00
01:00	11:00	1	1	0	99.73	99.62	100.00
02:00	12:00	1	1	0	99.86	99.81	100.00
09:00	19:00	1	1	0	100.00	100.00	100.00
00:00	10:00	0	0	0	100.00	100.00	100.00
04:00	14:00	0	0	0	100.00	100.00	100.00
16:00	00:00	0	0	0	100.00	100.00	100.00
18:00	02:00	0	0	0	100.00	100.00	100.00
23:00	09:00	0	0	0	100.00	100.00	100.00

Table 5-14 Difference in ambulance hours between real schedule and simplified schedules

Station	***3 Shift Cycle*** *Ambulance Hours*		***5 Shift Cycle*** *Ambulance Hours*	
	|fixed shifts - actual|	*(%)*	*|fixed shifts - actual|*	*(%)*
Chermside	14	3.33	0	0
Roma Street	31	10.3	15	4.98
Kedron Park	0	0	0	0
Greater Brisbane	496	6.78	204	2.79

5.3.2 Generating Incident Data

Incident arrivals, along with the requirements for each incident produced, are generated in this section.

5.3.2.1 Incident arrivals

Incident arrivals are generated from interarrival rates extracted from real arrival times for each priority type and sub priority category. A new incident of priority type p is generated at time t_p' from the formula $t_p' = t_p - \lambda_{p\tau}\ln(r)$ where

t_p = Time of last incident of priority type p

$\lambda_{p\tau}$ = Interarrival rate for incidents of priority type p during interval τ

τ = Time interval containing t_p

r = Random number distributed uniformly over (0,1)

An exponential distribution is used for incident arrivals. This is a common approach to deal with randomly arriving, independent incidents where the interarrival rate can be determined. The parameter $\lambda_{p\tau}$, measured in minutes, is extracted by grouping incidents into priority types for each hour of the week, and finding the average time between incidents, according to the duration of the time interval divided by the number of real incidents arising in that time. Incidents must also arrive at a location, described by a latitude and longitude for each incident. The area upon which it is desired to generate incidents has been limited to an area covering the five ambulance stations and two major hospitals selected for the case study and immediate surroundings. The latitude and longitude limitations are shown in Table 5-15. A spatial grid of 25 x 25 rectangles is then generated which covers the entire area of interest.

Table 5-15 Bounds on the area of interest for the case study

Position	Min (deg)	Max (deg)
Latitude	-27.475	-27.350
Longitude	153.00	153.10

Incidents are randomly distributed across the grid with probabilities extracted from incident density. Within each grid square, incidents are distributed randomly from a uniform 2D distribution. This process may allow incidents to arrive at locations that are not accessible by road. Travel times for incident data are approximated rather than extracted from a road network. This process is outlined in Section 5.3.3.

5.3.2.2 Hospital Transfer and Time Spent at Hospital

The new set of data to be generated requires parameters determining whether an incident requires transfer to a hospital and, if so, how long the processing of ramping, admission and cleaning will take before an ambulance is clear to respond to another incident. The probability of hospital transfer occurring for each priority category is extracted from the real data and used to randomly assign whether each newly generated incident will require transportation to a hospital.

Information for a collated processing time containing ramping times (waiting time at hospital), admitting a patient into the facility, and cleaning the ambulance is necessary where an incident is transferred to a hospital. For the deterministic data set, it is suitable to combine these values together as a single operation for 'processing time at hospital', because once an ambulance has arrived at a hospital, all these processes will follow on from each other at the same hospital, without any gaps in between activities. Incidents where no transfer to hospital is required will have a processing time of zero minutes. Any time required for cleaning would be recorded as time on scene for these instances.

Ramping times, in particular, are known to have a distribution where outliers with extreme ramping times of several hours can exist. A lognormal distribution is used to generate time spent at hospital to model processing times that may have extreme outliers from the mean value. Mean and standard deviation were extracted from the 2011/12 incident data as a function of priority categories and time, with four time intervals daily, each spanning six hours (12am-6am; 6am-12pm; 12pm-6pm; and 6pm to 12am). Values for these parameters are shown in Table 5-16. The lognormal probability distribution function is $f(x|\mu,\sigma) = \frac{1}{x\,\sigma\sqrt{2\pi}} e^{-\frac{(\ln x-\mu)^2}{2\sigma^2}}$.

Table 5-16 Lognormal distribution parameters for time spent at hospital

(Priority, Time Interval)	**μ**	**σ**
(1,1)	3.6169	0.8699
(1,2)	3.635	0.8987
(1,3)	3.7569	0.9682
(1,4)	3.6777	0.9851
(2,1)	3.4898	0.8951

(2,2)	3.5476	0.8678
(2,3)	3.6414	0.9787
(2,4)	3.5765	0.9711
(3,1)	2.901	1.3116
(3,2)	1.9141	1.5016
(3,3)	1.7696	1.6682
(3,4)	2.0584	1.8015

5.3.2.3 Ambulance Vehicle and Hospital Preferences

The data provided included information on the percentage of incidents requiring hospital transfer, but not on hospitals to which ambulances were directed for each incident. Assumptions are therefore required in order to test the ability of the model to select one hospital over another if required, or the best from multiple options if no preference is nominated. A summary of the assumptions is as follows:

- Incidents requiring transfer to hospital do not require hospitals outside of the area of interest
- Most, but not all, incidents requiring hospital transfer will not have a requirement for a specific hospital
- Incidents requiring a specific hospital are split evenly between the two hospitals in the area
- All priority codes have a probability of requiring a specific hospital
- Requirements for particular ambulance types are dependent on priority code

The case study assumes that every incident arising in the area is suitable to transfer to at least one of the two included hospitals where transfer is required. No hospitals outside the area were considered. The potential impact from this assumption is that travel times to hospitals may be shorter due to incidents where a patient who has a preference for a hospital outside the region is instead taken to a closer hospital. Additionally, an assumption introduced is such that the majority of incidents requiring transfer to a hospital can go to either hospital in the area. It is then assumed that the remaining incidents require a specific hospital out of the two available, with an equal percentage designed to each of the two hospitals in the region. The values are shown in Table 5-17.

Table 5-17 Percentage of incidents requiring transportation to hospital that are directed to specific hospitals

Hospital	*Percentage*
Either hospital	70%
1. PCH	15 %
2. RBWH	15%

This assumption is justified according to the following rationale. Hospital 1 in this case study, The Prince Charles Hospital (PCH), has a specialist unit for cardiac care while Hospital 2, The Royal Brisbane and Womens Hospital (RBWH) has specialist maternity services and is a designated major trauma centre. Wherever possible, emergency and urgent incidents with specialist requirements would be directed to a hospital with a matching specialist unit. Therefore, Code 1 and Code 2 incidents will have at least some cases where a particular hospital is preferred. Code 3 incidents contain a number of transportation incidents to particular hospitals for inter facility transfer or appointments. As such, a potentially larger number of Code 3 incidents will require a specific hospital but this information is unavailable for analysis. For simplicity, all priority codes have been assigned the same probability. Patient preference and previous treatment at a particular facility are also possible considerations for constraining the choice of hospital to which the patient should be transferred. These considerations can apply to any priority code but will not always be hard preferences. In order to allow only necessary restriction, and not enforce all preferences which may not be able to be met in the real world, the number of incidents allowed to travel to any hospital has been kept at a high percentage of 70%. Sensitivity of the model to hospital preferences requires additional data and is left for further work.

Vehicle requirements for each incident are also required for the models presented in this thesis. Three types of ambulance are assumed, as described in Table 5-18. The requirement for each vehicle type is based on the priority code of an incident, where more serious incidents require more qualified paramedics. The assumption made for this condition is that: Code 1 incidents require Type I ambulances 50% of the time and Type II ambulances the remaining 50%; Code 2 incidents require Type I ambulances 25% of the time, Type II ambulances 50% of the time and Type III ambulances only 25% of the time; and Code 3

ambulances require Type II ambulances 50% of the time and Type III ambulances 50% of the time. This approach allows the models to be tested for different ambulance types.

Table 5-18 The three different ambulance vehicle types considered in the case study.

Ambulance Type	***Relative Cost***	***Applicable incidents***
I	Most expensive and highest degree of training	All
II	Moderately expensive and moderate degree of training	Most
III	Least expensive and minimal degree of training	Least serious incidents only

5.3.2.4 Due Dates

Response targets for different priority types are extrapolated and simplified from QAS Response Code criteria. Code 1 incidents require immediate attention. Targets for the 50^{th} and 90^{th} percentiles are approximately 8 mins and 16 mins respectively. This is used to set tardy response time at 10 mins and an upper response time at 20 mins for generated incidents with priority code 1. Code 2 incidents require timely responses. The most severe of these require an undelayed response, while response targets for other subcategories are 30 mins to 60 mins. For the models presented in this thesis, the data sets a tardy response at 30 mins and an upper response limit at 60 mins for all urgent Code 2 incidents. Non-emergency incidents are time critical but routine transport where an ambulance is required is not time critical. As this model simplifies routine transport into the same category as all non-emergency incidents, all Code 3 incidents in the data are considered time critical. The tardy limit is set as 45 minutes (a more relaxed boundary than Code 1 or Code 2 incidents) with an upper limit of 2 hours set, as a delayed response time of greater than 2 hours would be unlikely to get patients to scheduled appointments at appropriate times. This information is summarised in Table 5-19 and applied to all generated incidents according to priority type.

Table 5-19 Priority Code response time targets

Priority Codes	**Tardy (mins)**	**Upper (mins)**
Code 1	10	20
Code 2	30	60
Code 3	45	120

5.3.3 Estimating Travel Times

The data set generated to test the models presented in this thesis requires information on estimated travel times between nodes visited by ambulances. Nodes include incident locations, ambulance stations, hospitals and, in the case of the real time model, the current location of an ambulance on the road. The static and dynamic models use deterministic data, while estimated travel times in the real time model may be generated or updated at trigger events. For a small number of locations known in advance, it is possible to manually enter data into free online tools to determine the expected travel time between two points using longitude and latitude. Google maps is used to determine travel times between hospitals and ambulance stations falling within the case study region under normal traffic conditions.

Unfortunately, the process of entering location data into an online engine becomes unwieldy for incident arrivals, due to the large number of incidents. Therefore, in order to quickly generate travel times, a simple program is developed (shown in Figure 5-19).

Algorithm 5.1 Travel Time Estimation for incident responses

1: Let (x_i,y_i) = Location of incident i & (x_j,y_j) = Secondary location

2: Generate random distance between two points based on straightest path

$$(a,b) = \frac{r_E \pi}{180} * 2 \operatorname{asin} \sin\left(\frac{|x_b - x_a|}{2}\right)^2 + \cos(x_a)\cos(x_b)\sqrt{\sin\left(\frac{|y_b - y_a|}{2}\right)^2}$$

3: Define average travel speed for initial response

$$V_p = \begin{cases} 80, & \text{if p} = 1, \text{(i. e. emergency incident and lights and sirens used)} \\ 60, & \text{otherwise} \end{cases}$$

4: Generate random travel time adjustment $r_t = \left(\frac{D(a,b)}{2} + 10\right) * r$

5: Estimate travel time between two points for initial response $T(a,b) = \frac{D}{V_p} + r_t$

6: Generate random travel time adjustment $r_t = \left(\frac{D(a,b)}{2} + 10\right) * r$

7: Estimate travel time between two points for all other times $T(a,b) = \frac{D}{60} + r_t$

Figure 5-19 Simple algorithm to generate estimated travel time between any two locations

This can be used to generate travel times quickly for the real time model as well as generating deterministic data. Assumptions for this algorithm include average travel times for ambulances under normal conditions and when using lights and sirens (for initial response to emergencies only) and assuming that travel time is the same in both directions. Expected variations in travel time due to time of day are not included in the simple travel time estimation.

The algorithm first estimates the straight line distance between two points and then estimates the travel time based on average speed over the distance. A small perturbation is applied to the travel time to reflect the fact that road networks are rarely straight lines between two points and travel speed will fluctuate around averages. The random travel time perturbation is a function of distance, as deviations from the straightest line will add more travel time for longer journeys, and will always be additive. A constant term appears in the perturbation function as well to randomise the effect of fluctuating travel speed.

This algorithm does not include modification for increased travel times during peak hour. This is a very simplistic method of estimating travel times quickly between any two points. The next section in this chapter discusses how well travel times from the formula match to exact travel times.

5.4 VERIFYING NEW DATA

A new data set is generated using methods described above. In this section, the new data set is compared to the real data. This investigates the quality of the data set used to solve the optimisation models developed in this thesis. The new data set covers only a two week period, compared against a year of actual data, and contains 2,758 incidents.

5.4.1 Incident Arrivals

Incident arrival times for the generated data for each hour of the week are shown in Figure 5-20. Expected daily seasonality is observed in the generated data (i.e. daily peaks and troughs on weekdays, weaker peaks and troughs over weekends). The density of incidents' arrival locations from the case study is overlaid onto a map of the Brisbane area in Figure 5-21, which also serves to highlight the size and scale of the area covered by the case study. To compare the spatial distribution of incidents from generated data to that of the real data, contour plots are produced (shown in Figure 5-22) and similar density patterns between the generated and real data circled. The density contours of the generated data are sparser, due to an order of magnitude fewer incidents in the data set. Similar patterns are observed between the real and generated data. Two major hotspots and several corridors of higher density are observed in each graph, while density gaps appear more smeared out in the generated data as a result of lower overall density.

5.4.2 Priority Type and Ambulance Vehicle Requirements

Table 5-20 shows the percentage of incidents of each priority type from the data set used in the case study. There is less than a 1% difference for each priority type between the new data set and the actual distribution of priority types. The assumptions made about ambulance vehicle requests for each priority type result in Type II ambulances being requested twice as often as Type I or Type III ambulances.

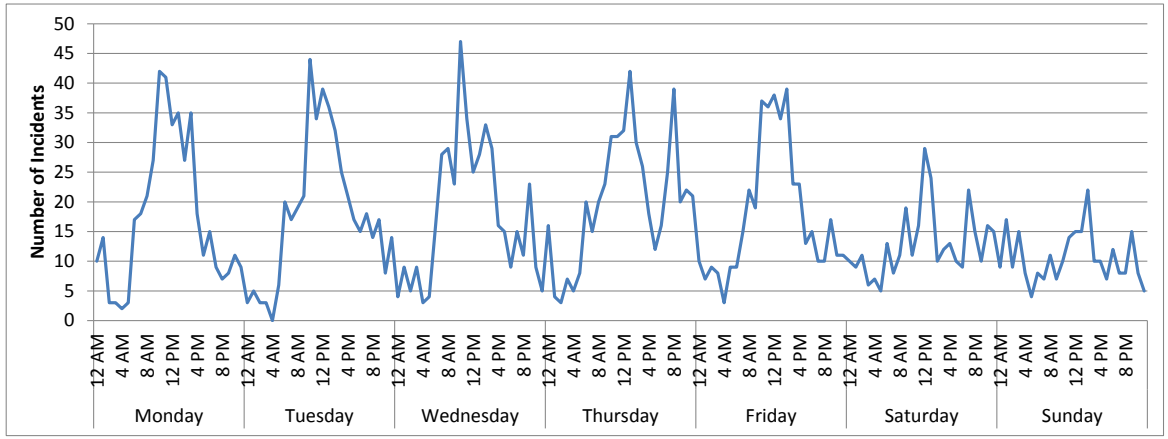


Figure 5-20 Incident arrival for two weeks of new generated data

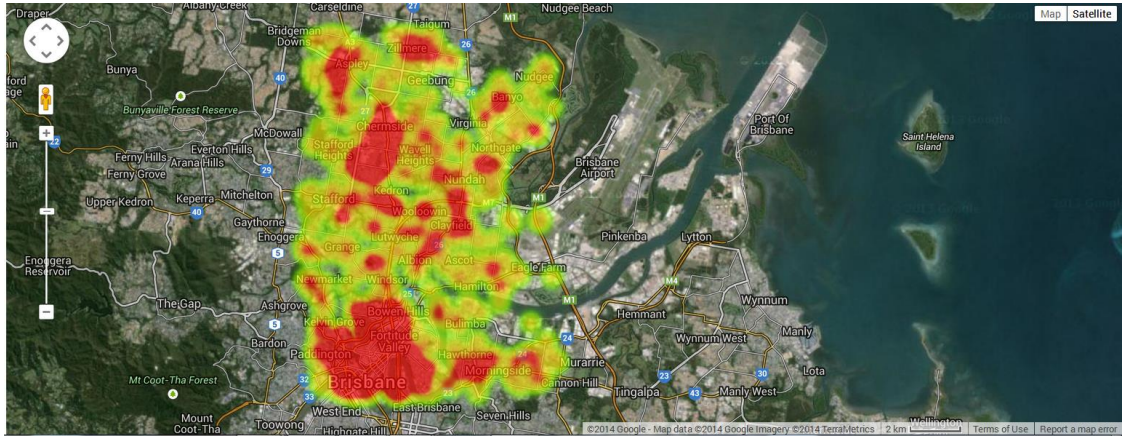


Figure 5-21 Incident location density of generated data set

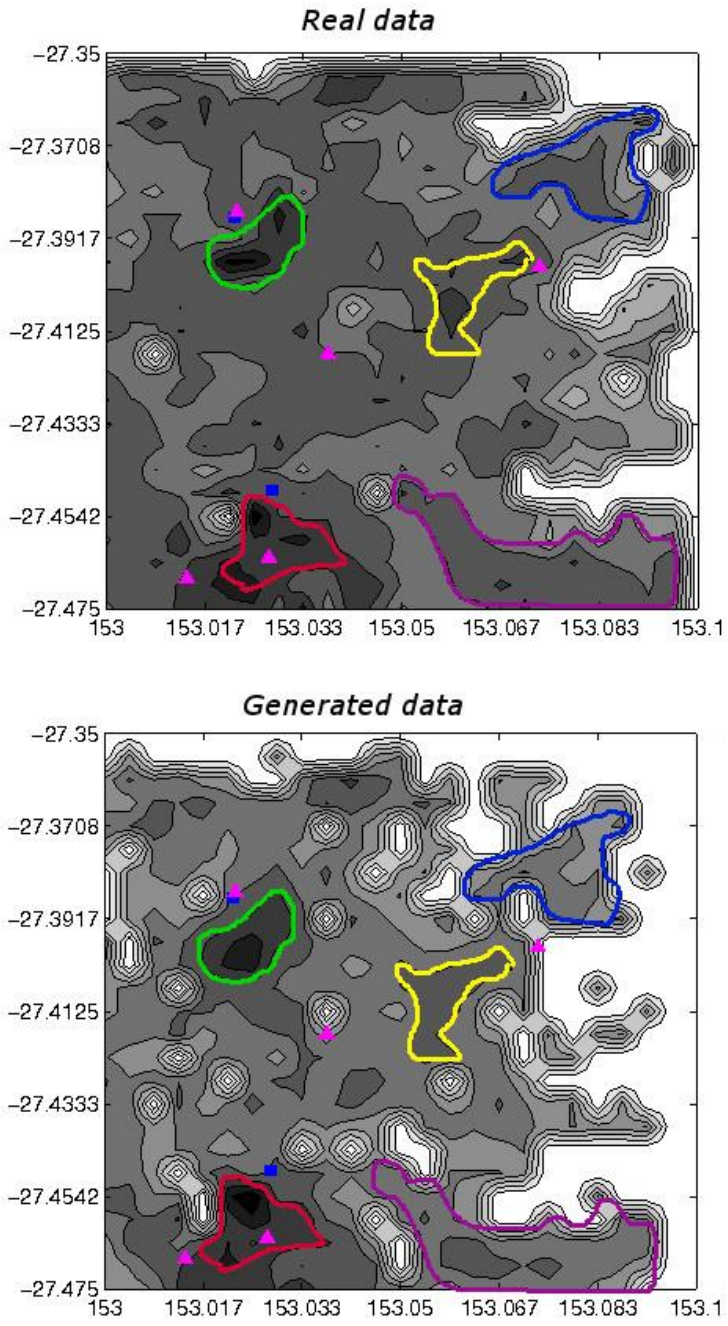


Figure 5-22 Percentage of incidents arising in a spatial grid representing the area of the case study

Table 5-20 Percentage incidents of each priority type and requested ambulances

Ambulance Type	Emergency Incidents	Urgent Incidents	Non-Emergency Incidents	All Incidents
Type I	16.68%	8.09%	0%	24.77%
Type II	16.24%	14.29%	19.47%	50.00%
Type III	0.00%	7.69%	17.55%	25.24%
All Types	32.92%	30.07%	37.02%	
2011/12 Incident Data	*33.82 %*	*29.35 %*	*36.83 %*	

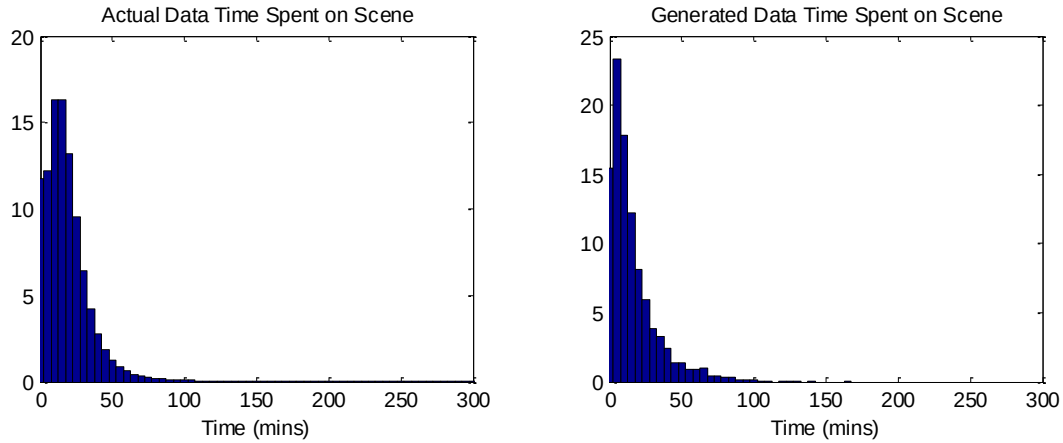


Figure 5-23 Distribution of the amount of time spent at the scene of incidents

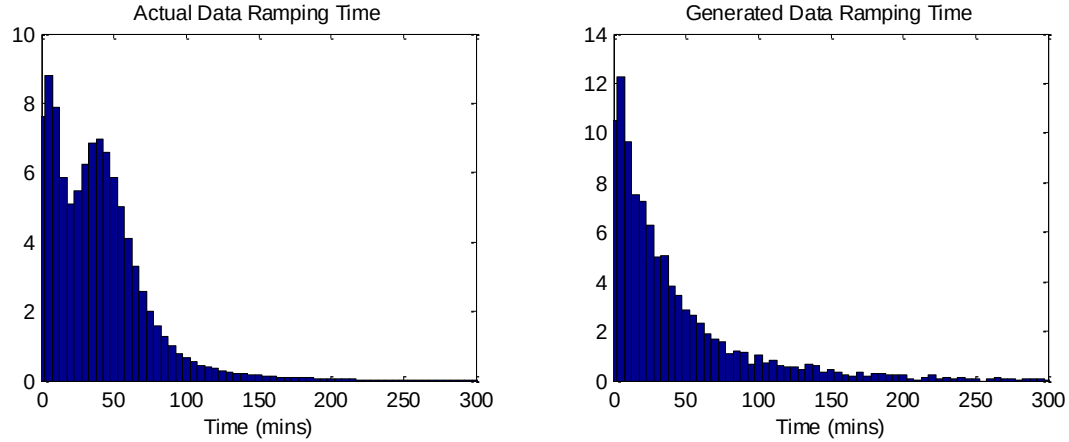


Figure 5-24 Distribution of time incidents spend waiting at hospitals

5.4.3 Time Spent at Incident Scene

The time that an ambulance spends at the scene of an incident has a distribution as shown in Figure 5-23. This figure indicates that the spread of time spent at scene in the generated data has a bias toward shorter processing times when compared with the actual data distribution. This is likely to affect the percentage of time that ambulances are busy in the final schedules. Possibilities for further work include additional analysis of the distribution of actual times spent at the scene to create a more accurate case study.

5.4.4 Hospital Transfers and Time Spent at Hospitals

Table 5-21 looks at the percentage of incidents requiring transportation to hospital. There are slightly fewer non-emergency incidents requiring hospital transportation in the generated data compared to the real data, but less than a 1% difference for all incidents. However, there were fewer incidents requiring hospital transfer in the 2011/2012 data than is recorded in public performance indicators for the following two years. It is possible that the method of extracting the rate of hospital transfers contains a flaw and the hospital transfer rate should be higher. This will also impact the percentage of time that ambulances are busy, and the locations at which they become available, in the final schedules.

Table 5-21 Percentage of incidents transferred to hospital

Priority	New Data Set	2011/12 Incident Data	2012/13 Public Data	2013/14 Public Data
All Incidents	80.75%	81.45%	86.8%	87.4%
Emergency Incidents	70.26%	71.04%		
Urgent Incidents	77.44%	77.08%		
Non-Emergency Incidents	92.75%	94.48%		

For incidents requiring transportation to hospital, the estimated time spent at the hospital is considered. Figure 5-24 compares time spent at hospital for the real data and the generated data. This figure shows that the distribution of time spent at hospital from the actual data contains two peaks, which is not matched in the generated data, which may be biased towards shorter times spent at hospital. However, the number of incidents able to be moved ‘off stretcher within 30 minutes’ in the generated data, at 56.25%, matches better with the observed ‘off stretcher within 30 minutes’ performance measure for September 2012 (Queensland Health,

2013). The latter measure is 56% for PCH, and for RWBH, is higher again than the 43.73% of incidents extracted from the 2011/12 incident data that were able to be moved off-stretcher in 30 minutes. Further work on developing a case study for the scheduling models in this thesis could focus on developing a bimodal distribution for times spent at hospital.

5.4.5 Travel Times

Estimated travel times are verified through comparison with samples of actual travel time. Three real incidents are selected and travel times to these locations from sample ambulance stations, under normal traffic conditions, extracted from Google Maps. The results are shown in Table 5-22. The average estimated travel time, calculated from 30 iterations, deviates no more than 4.1 minutes from the actual travel times in all of the examples tested. While not as accurate as results from a road network, this is an acceptable result for a simple method, able to return an estimate of travel time within milliseconds.

Table 5-22 Comparison of estimated and actual travel times for real incident locations

Location 1	**Location 2**	**Actual Travel Time from Google maps (mins)**	**Travel time from algorithm (mins)**		
			Average	Max	Min
Incident 1	Station A	15	13.63	19.46	9.15
Incident 2	Station A	9	10.37	16.74	5.02
Incident 3	Station A	18	22.12	29.88	13.89
Incident 1	Station B	8	7.37	12.53	2.16
Incident 2	Station B	12	14.23	21.13	7.78
Incident 3	Station B	33	31.85	41.70	21.82

Chapter 6: Static Model

The contribution of this chapter is to demonstrate that a single stage model for ambulance scheduling and crew scheduling is possible using Flexible Flow Shop Scheduling, and to provide initial results and analysis of the formulation. The initial model is a strategic model that always returns ambulances to their home station, so that overtime may be considered by calculating the difference in time between the end of a shift and the clear time of the last incident responded to by an ambulance during that shift.

This remainder of this chapter is set out as follows: Section 6.1, a description of the assumptions used to formulate the mathematical model and presentation of the model itself; Section 6.2, the proposed approach to solve the model with explanation of the heuristic algorithms employed; Section 6.3, the results from the model and a discussion on the sensitivity of the model to problem size and objective weights; Section 6.4, additional variations explored for the mathematical model; Section 6.5, implications of the proposed model and suggested further work.

6.1 FORMULATION

In this section, the formulation of a mathematical model using deterministic data is discussed. The problem is formulated using Flexible Flow Shop Scheduling (FFSS) techniques. The model assigns ambulance vehicles to stations and shifts and meets demand by directly assigning vehicles to incidents. Shift scheduling rules for ambulance crews are included as constraints under the assumption that the same team of staff work as a unit each time they are scheduled to work a shift. An overview of the model is presented in Figure 6-1. The rest of this section explains the assumptions and presents the mathematical model.

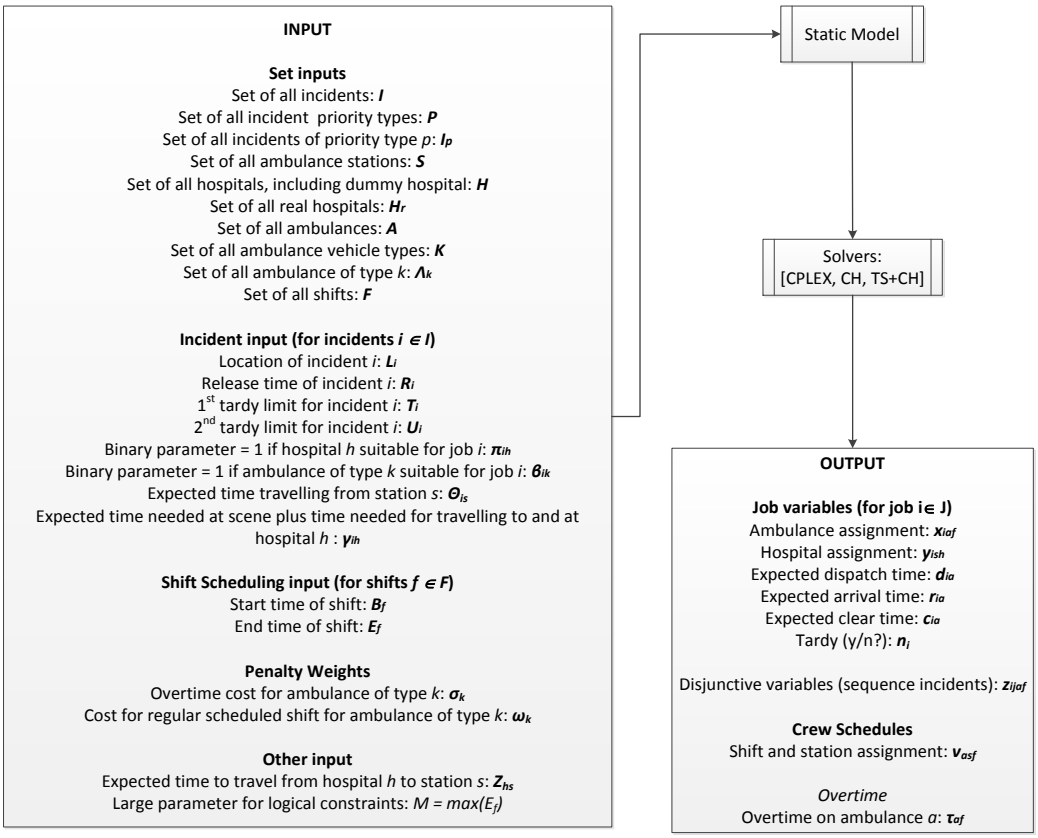


Figure 6-1 Overview of the static model input, solution approach and output

6.1.1 Assumptions

A number of assumptions are present in this modeland explained in this section. For ease of understanding, the assumptions are listed here and further discussed below:

- Availability of ambulances is the availability of ambulance crew (not the ambulance vehicle)
- Different crew mixes have different costs and are able to respond to different sub-sets of all incidents
- Each incident is a job which requires five operations to be completed after dispatch (see Figure 6-1).
- All time spent at a hospital is gathered into a single operation
- Precedence relationships between operations must be obeyed
- Pre-emption is not permitted

- Ambulance crew may be scheduled onto multiple shifts
- There must be a minimum eight hour break between shifts
- Running costs consist of costs for scheduled shifts and overtime
- Overtime is not limited
- Overtime costs double the per-minute cost of regular time
- Overtime is paid by the minute, not in blocks
- Ambulances must return to their home ambulance station
- Ambulances cannot respond to new incidents before they start a shift or after the time they were due to end a shift
- Overtime is accrued if, and only if, ambulances are not available at their home ambulance station at the designated end of their shift
- Incidents receive exactly one ambulance
- All ambulances are capable of transferring patients to hospital
- Hospital preferences must be met under all circumstances
- Ramping time at a hospital is independent of the number of patients arriving at the hospital by ambulance
- A response is tardy if an ambulance does not arrive by the first due date specified in Section 5.3.2.4
- Tardy responses are limited to a given percentage of all incidents for each priority type, based on performance targets for ambulance services
- All incidents must receive a response by the second due date specified in Section 5.3.2.4

For the ambulance problem, there exists a fleet of ambulance vehicles able to act as alternative machines for processing tasks. An ambulance vehicle requires an ambulance crew in order to be dispatched to respond to an incident. Ambulance availability is limited by the available ambulance crew rather than the vehicle itself. There are different types of vehicles with different crew mixes that are able to respond to a sub-set of all incidents. For simplicity, where this model refers to an

ambulance it refers to any ambulance vehicle with an appropriate ambulance crew. Ambulance IDs refer to the ambulance crew and not the vehicle.

Each incident is considered as a job in a FFSS problem and passes through five main operations. These are: travelling to the scene of an incident; treatment at the scene; travelling to a hospital; admission to a hospital; and return travel to the ambulance station. Figure 6-2 shows the interaction of these processes. Preparations to respond to or depart from the incident scene are included in travel time to the scene and treatment time at the scene respectively. Ramping (the time that an ambulance spends at the hospital in a queue waiting to admit a patient) is placed together with the actual time taken to admit the patient into the care of the hospital. This is done to simplify the model and focus on total 'time to clear'. Strict precedence relations exist between the operations and pre-emption is not allowed.

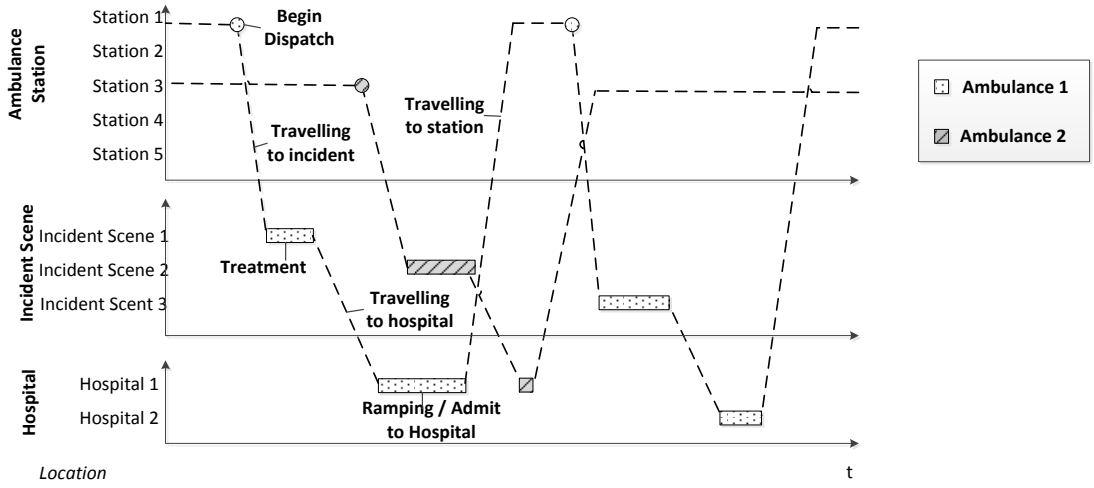


Figure 6-2 Example of schematic representation of the ambulance processes involved for the static model

Shift schedules for ambulance crews are built around required dispatch events. Whenever the shift schedule re-uses an ambulance on a new shift, this refers to the same ambulance crew being assigned a new shift. This is a simplifying assumption of real life where shift schedules are created that will ensure an appropriate crew mix on each ambulance but may or may not consist of the same individuals each time. The benefit of the simplifying assumption is that shift scheduling rules for ambulance crew can be integrated into a single model as shift scheduling constraints.

Shift scheduling rules surrounding fatigue breaks are integrated into the scheduling model to explore the concept of integrating the model. Workplace agreements state that ambulance employees require 'an uninterrupted break from duty of at least 8 consecutive hours between the cessation of a scheduled shift and the commencement of the following roster' (Queensland Industrial Relations Commission, 2012, p.327). A preference for forward rostering also exists. Forward rotation of shifts is a policy where the next shift begins later in the day than the prior shift and has been recommended to combat fatigue.

Shift scheduling costs are integrated with overtime costs in the objective function. This approach has been selected for the static model as it is a novel way to investigate the costs associating with running ambulance services while maintaining commitment to performance targets.

Overtime, that is, time worked outside of normal hours, requires the ambulance crew to be paid at a higher rate. For the purpose of this model, only overtime that is accrued at the end of a normal shift is considered and ambulance crews cannot be recalled once they have left the ambulance station. Ambulance crews are assumed to have left the station once they have returned home and the due end time of the shift has passed. Overtime penalty rates applied to ambulance crews called in to work additional shifts are not considered in this model, which creates the schedule. Penalty rates vary from between 1.5 and 2 times the standard rate, plus additional meal allowance penalties, depending on duration of overtime and when it occurs. To simplify the model, a standard double rate time is used. No minimum or maximum limit on overtime has been applied. Overtime costs are applied on a per-minute basis to minimise the number of minutes used.

To allow overtime to be calculated, it is assumed that ambulances must always return to their home station before being considered clear to either finish a shift or begin another job. In real life, ambulances may be dispatched to new jobs before returning to a station, therefore, this model is expected to find an upper bound on the minimum number of ambulance crew shifts required rather than the optimal number.

The static model in this chapter has constraints to guarantee that each incident receives exactly one ambulance. In real life, mass casualty events can occur that would require more than one ambulance. The static model instead uses the

simplifying assumption that real incidents requiring multiple ambulances will be considered as multiple incidents within the model. Further simplifying assumptions in the static model are the use of hard constraints for selection of hospitals and the assumption that all ambulances are equipped to transport patients to a hospital. In reality, the selection of hospital to which a patient is transferred may have subtle preferences among the options available. This variation is not used because including soft constraints for hospital preferences would require penalty costs to be added into the objective function, and such costs are not directly related to the costs of weighted ambulance hours for shifts and overtime. It is considered sufficient that hospitals and ambulances that are considered unsuitable will be excluded. For simplicity, the ramping time at hospitals is not dependent on the number of patients already sent to the hospital. The relationship between number of patients and ramping time is considered out of scope for this model.

6.1.2 Parameters

The following parameters are used in the model:

F Set of shifts

P Set of priority types

I Set of all incidents

I_p Set of incidents of priority type p

H Set of hospitals including a dummy hospital to represent incidents not requiring transfer

H^r A subset of H introduced to represent only real hospitals

K Set of ambulance types

A_k Set of ambulances of type k

B_f Beginning time of shift f

E_f Ending time of shift f

N_p Maximum number of tardy arrivals allowed for patients of priority type p

R_i Ready time of incident i (i.e. clock time that a call has been assessed and is able to be assigned an appropriate ambulance)

D_i Recommended arrival time of incident i (clock time)

U_i Upper limit for arrival time of incident i (clock time)

$\beta_{ik} = \begin{cases} 1, & \text{if ambulance type } k \text{ is able to respond to incident } i \\ 0, & \text{otherwise} \end{cases}$

$\gamma_{ih} = \begin{cases} 1, & \text{if hospital } h \text{ is able to receive patient from incident } i \\ 0, & \text{otherwise} \end{cases}$

θ_{is} Expected travel time to incident scene *i* from ambulance station *s*

ψ_{ih} Expected time for incident to clear at hospital *h* (i.e. expected stabilisation time for incident *i* plus expected travel time from the incident scene of incident *i* to hospital *h*, plus expected offload time for incident *i* at hospital *h*)

ζ_{sh} Expected travel time from hospital *h* to ambulance station *s*

ω_k Cost of an ambulance of type *k* for one shift

σ_k Cost of one unit of overtime for an ambulance of type *k*

M A large value for use in logical constraints with value $\max_{f \in F}(E_f)$

6.1.3 Variables

6.1.3.1 Decision Variables

Three binary decision variables are introduced to handle the assignment, sequencing and scheduling of resources and operations. These indicate which ambulance responds to each incident and when, the hospital to which an incident was transferred, and where and when ambulances were assigned to work.

$$x_{iaf} = \begin{cases} 1, \text{if ambulance } a \text{ is dispatched to incident } i \text{ during shift } f \\ 0, \text{otherwise} \end{cases}$$

$$y_{ish} = \begin{cases} 1, \text{if ambulance travels from station } s \text{ to incident } i \text{ and hospital } h \\ 0, \text{otherwise} \end{cases}$$

$$v_{asf} = \begin{cases} 1, \text{if ambulance } a \text{ assigned to ambulance station } s \text{ during shift } f \\ 0, \text{otherwise} \end{cases}$$

6.1.3.2 Dependent Variables

The following variables are also determined by the model

d_{ia} Dispatch time for dispatch of ambulance a to incident i

r_{ia} Arrival time of ambulance a incident site i

c_{ia} Clear time of incident i on ambulance a

o_{af} Overtime accrued on ambulance a at the end of shift f

$n_i = \begin{cases} 1, \text{if incident } i \text{ received a tardy arrival} \\ 0, \text{ otherwise} \end{cases}$

$z_{ija} = \begin{cases} 1, \text{if the start time of incident } i \text{ on ambulance } a \text{ is later than or} \\ \quad \text{equal to the clear time of incident } j \text{ on ambulance } a \\ 0, \text{ otherwise} \end{cases}$

6.1.4 Objective

The objective function minimises the costs of running the ambulance services, considering different types of ambulances (with appropriate ambulance crews), assigned shifts and overtime.

Minimise

$$\sum_{k \in K} \left(\omega_k \sum_{a \in A_k} \sum_{s \in S} \sum_{f \in F} v_{asf} + \sigma_k \sum_{a \in A_k} \sum_{f \in F} o_{af} \right)$$

6.1.5 Constraints

Precedence constraints

Precedence constraints are introduced to ensure that the operations for each incident occur in the appropriate order at the appropriate time and apply the correct processing time.

Constraint (6.1): The earliest clock time that an ambulance is able to be dispatched to an incident must be greater than the release date for that incident:

$$d_{ia} + M \left(1 - \sum_{f \in F} x_{iaf} \right) \geq R_i \quad \forall\, i \in I, a \in A \tag{6.1}$$

Constraint (6.2): The arrival time (i.e. clock time that the ambulance arrives on the scene of an incident) must be greater than the dispatch time plus correct travel time:

$$r_{ia} + M \left(1 - \sum_{f \in F} x_{iaf} \right) \geq d_{ia} + \sum_{s \in S} \sum_{h \in H} \theta_{is} y_{ish} \quad \forall\, i \in I, a \in A \tag{6.2}$$

Constraint (6.3): The clock time that an ambulance clears an incident must be the arrival time plus necessary travel time and time at hospital:

$$c_{ia} + M\left(1 - \sum_{f \in F} x_{iaf}\right) \geq r_{ia} + \sum_{s \in S} \sum_{h \in H} (\psi_{ih} + \zeta_{sh}) y_{ish} \quad \forall\, i \in I, a \in A \qquad (6.3)$$

Constraints (6.4), (6.5) and (6.6): Dispatch, arrival and clear time for incident *i* on ambulance *a* must be zero if ambulance *a* is not assigned to incident *i*. These constraints reduce the size of the feasible solution space without affecting the optimal solution:

$$d_{ia} \leq M \sum_{f \in F} x_{iaf} \quad \forall\, i \in I, a \in A \qquad (6.4)$$

$$r_{ia} \leq M \sum_{f \in F} x_{iaf} \quad \forall\, i \in I, a \in A \qquad (6.5)$$

$$c_{ia} \leq M \sum_{f \in F} x_{iaf} \quad \forall\, i \in I, a \in A \qquad (6.6)$$

Disjunctive Constraints

The binary variable z_{ija} handles disjunctive constraints that sequence, without overlap, any two incidents using the same ambulance.

Constraints (6.7) and (6.8): Paired disjunctive constraints are introduced to ensure that the clear time of the first incident is less than or equal to the dispatch time of the second incident:

$$d_{ia} - c_{ja} \geq M(z_{ija} - 1) \quad \forall\, i \in I, j \in I \setminus \{i\}, a \in A \qquad (6.7)$$

$$d_{ja} - c_{ia} \geq -Mz_{ija} \quad \forall\, i \in I, j \in I \setminus \{i\}, a \in A \qquad (6.8)$$

Tardy response constraints

An upper bound limits the clock time at which an ambulance should have reached the scene of an incident.

Constraint (6.9): Prevent responses from arriving unacceptably late:

$$r_{ia} \leq U_i \quad \forall\, i \in I, a \in A \qquad (6.9)$$

Constraint (6.10): Determine whether an incident received a tardy response, within allowable tardy limits, based on the clock time at which it was desirable to receive an ambulance on scene and the time at which an ambulance actually arrived. The variable n_i and this constraint are necessary to limit the total number of tardy incidents in the next constraint.

$$M\, n_i \;\geq r_{ia} - D_i \qquad \forall\, i \in I, a \in A \tag{6.10}$$

Constraint (6.11): Limits the number of incidents of each priority type that are allowed to have tardy responses in order to meet performance requirements:

$$\sum_{i \in I_p} n_i \;\leq N_p \qquad \forall\, p \in P \tag{6.11}$$

Overtime Constraints

Overtime constraints are a novel contribution of this model.

Constraint (6.12): The overtime decision variable o_{af} has a value greater than or equal to the time that ambulance *a* is active after it was due to end shift *f*. If all incidents are cleared prior to the end of the shift, overtime will be equal to zero:

$$c_{ia} - \; o_{af} - \; E_f \;\leq M\left(1 - x_{iaf}\right) \qquad \forall\, i \in I, a \in A, f \in F \tag{6.12}$$

Shift Scheduling Constraints

Another contribution of this formulation is to include ambulance crew shift scheduling rules directly into an ambulance scheduling model. This process has initially been tested with rules governing time off between shifts for ambulance crews; that is, business rules in the ambulance environment require an eight hour 'fatigue break' between shifts. These constraints apply to ambulance crews, which are not required to use the same ambulance vehicle each shift as long as another appropriate vehicle is available. Ambulance vehicles may be utilised by a number of different ambulance crews. In the model, the set of ambulances refers to the set of ambulance crews placed on appropriate ambulance vehicles.

Constraint (6.13): There must be a minimum of two shifts off between any two shifts on which an ambulance is scheduled. This formulation works well for fixed shift patterns where shifts are of equal duration:

$$\sum_{s \in S} v_{asf} + \sum_{s \in S} v_{as'(f+1)} + \sum_{s \in S} v_{as''(f+2)} \leq 1 \quad \forall\, a \in A, f \in F \tag{6.13}$$

Resource and Availability Constraints

Constraint (6.14): Only ambulances with a suitable ambulance crew and vehicle type can respond to an incident:

$$\sum_{f \in F} \sum_{a \in A_k} x_{iaf} \leq \beta_{ik} \quad \forall\, i \in I, k \in K \tag{6.14}$$

Constraint (6.15): Each incident receives exactly one ambulance and can only start during a single shift:

$$\sum_{a \in A} \sum_{f \in F} x_{iaf} = 1 \quad \forall\, i \in I \tag{6.15}$$

Constraints (6.16) and (6.17): Ambulances cannot be dispatched to an incident either before the start, or after the designated end, of the shift to which the incident is assigned. It is allowed for an ambulance to continue dealing with an incident past the end of the shift if they were assigned before the end of the shift. These paired constraints are necessary to ensure that ambulances are only dispatched during a shift that they are scheduled to work:

$$d_{ia} - E_f \leq M\left(1 - x_{iaf}\right) \quad \forall\, i \in I, a \in A, f \in F \tag{6.16}$$

$$B_f - d_{ia} \leq M\left(1 - x_{iaf}\right) \quad \forall\, i \in I, a \in A, f \in F \tag{6.17}$$

Constraint (6.18): Ambulances may be dispatched from only one ambulance station per shift. This simplification is necessary to ensure that ambulances in the static model are always dispatched from, and return to, their home ambulance station:

$$\sum_{s \in S} v_{asf} \leq 1 \quad \forall\, a \in A, f \in F \tag{6.18}$$

Each dispatched ambulance travels from a single ambulance station, is directed to exactly one hospital (real or 'dummy') and returns to the same ambulance station after each incident. It is also necessary to provide constraints ensuring patients receive treatment at the appropriate facilities.

Constraint (6.19): Each incident must be directed to a hospital and associated with a single ambulance station:

$$\sum_{s\in S}\sum_{h\in H} y_{ish} = 1 \qquad \forall\, i \in I \tag{6.19}$$

Constraint (6.20): Patients will be transferred to a real hospital if hospital transfer is required:

$$\sum_{h\in H^r}\sum_{s\in S} y_{ish} \geq \max_{h\in H^r} \gamma_{ih} \qquad \forall\, i \in I \tag{6.20}$$

Constraint (6.21): Patients cannot be transferred to inappropriate hospitals whenever there is a preference for a particular hospital or hospitals:

$$\sum_{s\in S} y_{ish} \leq \gamma_{ih} \qquad \forall\, i \in I, h \in H \tag{6.21}$$

Constraint (6.22): Define a relationship between the v_{asf}, x_{iaf} and y_{ish}variables. This constraint makes sure that if an ambulance is dispatched to an incident then the path travelled by that ambulance is from the correct ambulance station for that ambulance, to the incident site, to the correct hospital for that incident and back to the same ambulance station:

$$v_{asf} \geq x_{iaf} + y_{ish} - 1 \qquad \forall\, i \in I, a \in A, s \in S, h \in H, f \in F \tag{6.22}$$

Symmetry Breaking Constraints

Constraint (6.23): Force ambulances with a lower index to be selected first where multiple ambulances of the same type exist. This restricts duplication of solutions and improves the solution time:

$$\sum_{s\in S}\sum_{f\in F} v_{a^1sf} \leq \sum_{s\in S}\sum_{f\in F} v_{a^2sf} \qquad \forall k \in K, a^1 \in A_k, a^2 \in A_k : (a^2 = a^1 + 1) \tag{6.23}$$

Constraints (6.24) and (6.25): Non-negativity and integer constraints are also required for the model:

$$0 \leq d_{ia}, r_{ia}, c_{ia}, o_{af} \leq M \qquad \forall\, i \in I, a \in A \tag{6.24}$$

$$x_{iaf}, y_{ish}, v_{asf}, z_{ija}, n_i \in \{0,1\} \qquad \forall\, i \in I, a \in A, s \in S, h \in H, f \in F \tag{6.25}$$

6.2 SOLUTION APPROACH

The static model is solved for the case study discussed in Chapter 5. Solutions for the model are obtained from a MIP solver, basic Constructive Heuristic (CH) and a Tabu Search (TS) metaheuristic hybridised with a CH.

6.2.1 Case Study

Results for the static model are all obtained using a single case study. This allows the model to be solved repeatedly under the same conditions. This is of benefit when investigating multiple solution approaches that should be compared using the same set of data. It is also of benefit for exploring variability in results from non-exact solution methods. However, there are also risks with using a single scenario. An optimal schedule for one data set is not guaranteed to be optimal for another data set. The integrated ambulance scheduling and ambulance crew scheduling model uses the data set in the case study to determine the locations at which ambulances are required at different times. This directly informs an ambulance crew shift schedule. While the ambulance schedule is dependent on individual incidents in the case study, the ambulance crew schedule is more robust. Each time an ambulance crew is scheduled to a shift, they are able to respond to multiple scenarios. This lessens the impact of individual incidents.

6.2.2 Constructive Heuristic

The CH uses a greedy First Come First Serve (FCFS) algorithm to determine the number of each type of ambulance crew required to meet the demand without any avoidable tardy responses. Information about incidents is considered in order of arrival and, if possible, an ambulance crew from the pool of already assigned ambulance crews is assigned before any new ambulances are introduced. The process diagram for the CH is presented in Figure 6-3.

The process investigates possible ambulance assignments for each incident in the order in which they arrive. Ambulance crews are only added into the system when a new ambulance crew is needed. For each incident, feasible ambulance and hospital assignments based on incident requirements are identified. Ambulances are then tested in order of earliest arrival time. If an ambulance being trialled for assignment to an incident is unable to be scheduled onto a feasible shift it will be excluded and the next ambulance on the list tested. Ambulances are also excluded if

other incidents, which were previously assigned on the ambulance, overlap with all possible non-tardy responses for the current incident. In the event that all feasible ambulances are tested without an assignment being accepted, a new ambulance is introduced and assigned to the current incident with the earliest possible arrival time and cheapest ambulance type. This ambulance is then available for scheduling later incidents. The algorithm for the CH and its sub processes are shown in Figure 6-4, Figure 6-5 and Figure 6-6.

The CH finds feasible solutions but is not guaranteed to find optimal solutions. The process is restricted by not accepting any avoidable tardy responses at all, not allowing minor delays which would allow an incident to receive a response from an ambulance beginning a shift instead of an ambulance near the end of a shift, and by assigning the most basic allowable ambulance type when a new vehicle is assigned rather than considering which ambulances may be required later. These issues may be addressed by adding parameters into the CH addressing the probability or accepting tardiness, different dispatching locations and different ambulance types. This is addressed in the CH used for hybrid heuristics in the next section.

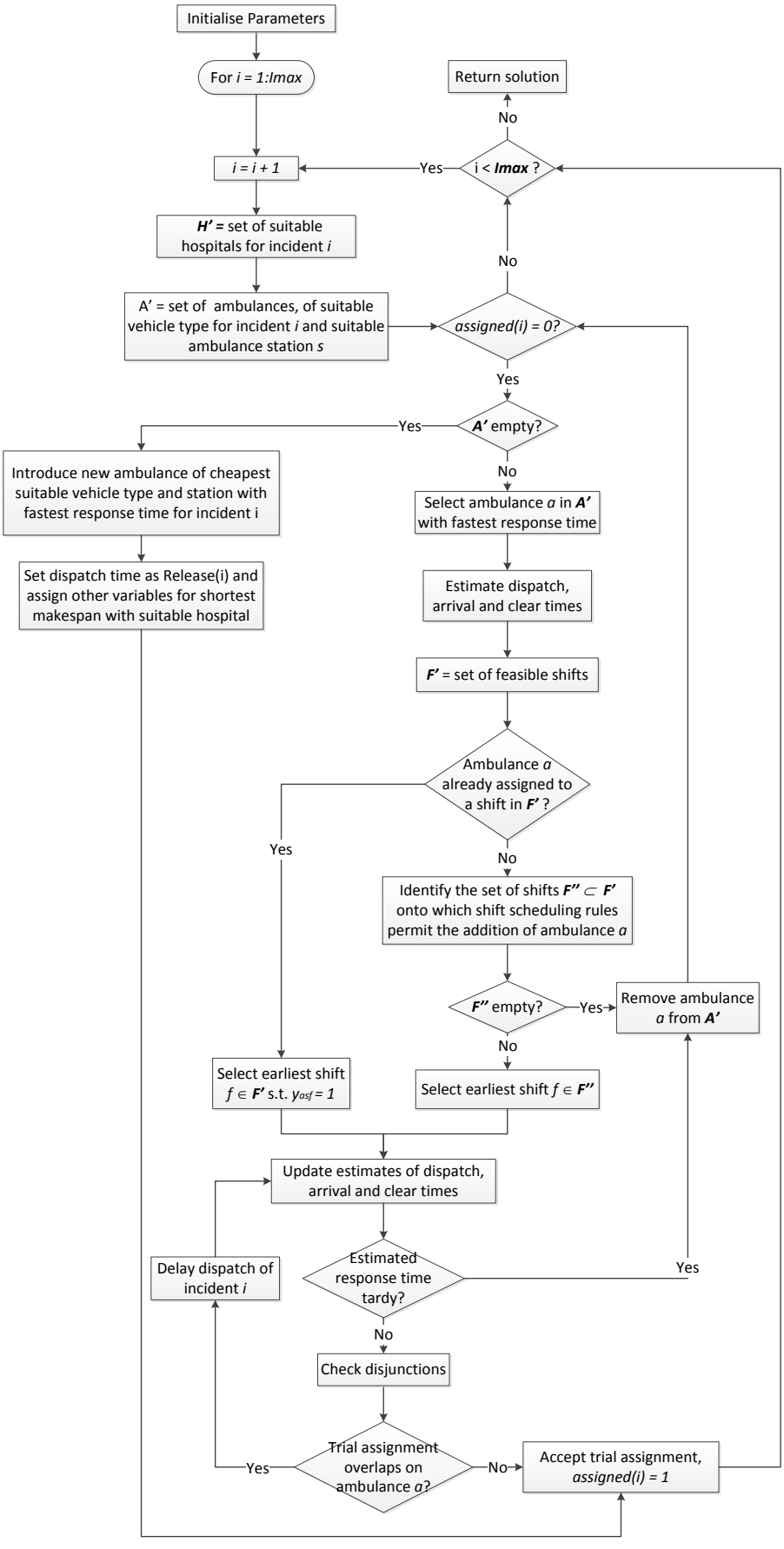


Figure 6-3 Process diagram for basic FCFS

Basic FCFS Constructive Heuristic

For *i= 1 to* I_{max}
  Select $\boldsymbol{F'} \subset \boldsymbol{F}$ s.t. $B_{\boldsymbol{F}'} \leq U_j$ && $E_{\boldsymbol{F}'} \geq R_j$
  Select $\boldsymbol{A'} \subset \boldsymbol{A}$ s.t. $\xi_{i\boldsymbol{A}'} = 1$ && $\left(\sum_{f \in \boldsymbol{F}} \boldsymbol{z}_{\boldsymbol{A}' sf}\right) \times D_i \geq R_i + \theta_{is}$
  Select $\boldsymbol{H'} \subset \boldsymbol{H}$ s.t. $\gamma_{\boldsymbol{iH}'} = 1$
  Set *assigned = 0*
  While *assigned = 0*
    If *isempty(**A'**)*
      Run Assign_New
      assigned = 1
    ElseIf $\min_{a \in \boldsymbol{A}'}\left(\max_{\mathrm{f} \in \mathbf{F}} \sum_{s \in \boldsymbol{S}}\left(z_{\boldsymbol{A}' sf} \times \theta_{is}\right) > D_i - R_i\right)$
      Run Assign_New
      assigned = 1
    Else
      $a = \arg\left(\min_{\mathrm{a} \in \mathbf{A}'}\left(\max_{\mathrm{f} \in \mathbf{F}} \sum_{s \in \boldsymbol{S}}\left(z_{\boldsymbol{A}' sf} \times \theta_{is}\right)\right)\right)$
      $s = \arg\left(\max_{s \in \boldsymbol{S}}\left(\max_{\mathrm{f} \in \mathbf{F}} z_{asf}\right)\right)$
      If $\left(\sum_{f \in \boldsymbol{F}'} z_{asf}\right) > 0$
        $f = \arg\left(\min_{f \in \boldsymbol{F}'}\left(B_f + M(1 - \mathrm{z}_{\mathrm{asf}})\right)\right)$
      Else
        $\boldsymbol{F}_{\boldsymbol{block}} = \{f-2, f-1, f, f+1, f+2\}$
          $\forall f \in \boldsymbol{F}$ s.t. $z_{asf} = 1$
        $\boldsymbol{F}'' = \boldsymbol{F}' \setminus \boldsymbol{F}_{\boldsymbol{block}}$
        If *isempty(**F''**)*
          f = 0
          $\boldsymbol{A}' = \boldsymbol{A}' \setminus a$
        Else *f = **F''**(1)*
        End If
      End If
      If *f > 0*
        $d = \max(R_i, B_f)$
        $r = d + \theta_{is}$
        $c = r + \min_{\mathrm{h} \in \mathbf{H}}(\psi_{ih} + \zeta_{sh})$
        If $r > D_i$
          $\boldsymbol{A}' = \boldsymbol{A}' \setminus a$
        Else *(d,r,c)* = Disj_Check*(d,r,c,i,a,f,s,h)*
          If $d > B_f$
            $\boldsymbol{A}' = \boldsymbol{A}' \setminus a$
          ElseIf $r > D_i$
            $\boldsymbol{A}' = \boldsymbol{A}' \setminus a$
            Else Save selected path
            $x_{iaf} = 1, y_{ish} = 1, z_{asf} = 1,$
            $d_{ia} = d, r_{ia} = r, c_{ia} = c, n_i = 0,$
            $o_{af} = \max\left(o_{af}, c_{ia} - E_f\right), assigned = 1$
          End If
        End If
      End If
    End If
  End While
End For
Return $\sum_{k \in \boldsymbol{K}}(\omega_k \sum_{a \in \boldsymbol{A_k}} \sum_{s \in \boldsymbol{S}} \sum_{f \in \boldsymbol{F}} z_{asf} + \sigma_k \sum_{a \in \boldsymbol{A_k}} \sum_{f \in \boldsymbol{F}} o_{af}$

Figure 6-4 Algorithm for basic constructive heuristic

Assign_New

Select cheapest ambulance $k = \arg(\min_{k \in \boldsymbol{K}'} (\omega_{k'}))$ where $\boldsymbol{K}' \in \boldsymbol{K}$ s.t. $\beta_{i\boldsymbol{K}'} > 0$
Add new ambulance a of type k ($\boldsymbol{A_k} = \{\boldsymbol{A_k}, a\}$)
$d_{ia} = R_i$
Select $f = \arg\left(\max_{f \in \boldsymbol{F}'} (B_f)\right)$ where $\boldsymbol{F}' \subset \boldsymbol{F}\ s.t\ B_{\boldsymbol{F}'} < R_i$
Select station with fastest response $s = \arg(\min_{s \in \boldsymbol{S}} (\theta_{is}))$
Select hospital $h = \arg(\min_{\mathrm{h} \in \mathbf{H}'} (\psi_{ih} + \zeta_{sh}))$ where $\boldsymbol{H}' \in \boldsymbol{H}$ s.t. $\gamma_{i\boldsymbol{H}'} > 0$
Set variables $x_{iaf} = 1,\ y_{ish} = 1,\ z_{asf} = 1, q_{ija} = 0\ \forall j \in \boldsymbol{I}$,
$r_{ia} = d_{ia} + \theta_{is},\ c_{ia} = r_{ia} + \psi_{ih} + \zeta_{sh}$
If $r_{ia} > D_i$
    $n_i = 1$
Else $n_i = 0$
End If
$o_{af} = \max(0, c_{ia} - E_f)$

Figure 6-5 Algorithm for assigning new ambulances in the static model

Disjunctive_check

Select $\boldsymbol{J} \subset \boldsymbol{I} \backslash \{i\}$ s.t. $x_{jaf} = 1$
If *~isempty(**J**)*
    Set *test_complete = 0*
    While *test_complete)< 0*
        $q_{i\boldsymbol{I}a} = 0$
        $\boldsymbol{J}' \subset \boldsymbol{J}$ s.t. $d_{\boldsymbol{J}'a} \leq d$ (precedents)
        If *~isempty(**J'**)*
            $q_{\boldsymbol{J}'ia} = 1$
            $j = \arg\left(\max_{j \in \boldsymbol{J}'} (c_{ja})\right)$
            $d = \max(d, c_{ja}), r = d + \theta_{is},\ c = r + \psi_{ih} + \zeta_{sh}$
        End If
        test_complete= 1
        $\boldsymbol{J}' \subset \boldsymbol{J}$ s.t. $d_{j_1 a} \geq d$ (antecedents)
        If *~isempty(**J'**)*
            $j = \arg\left(\min_{j \in \boldsymbol{J}'} (d_{ja})\right)$
            If $d_{ja} < c$
                $d = c_{ja}, r = d + \theta_{is},\ c = r + \psi_{ih} + \zeta_{sh}$
                test_complete= 0
            Else $q_{i\boldsymbol{J}'a} = 1$
            End If
        End If
    End While
End If

Figure 6-6 Sub-function to ensure disjunctive constraints are met in the static model.

6.2.3 Hybrid Heuristic

Each solution contains a sequence in which incidents are assigned to ambulances. This sequence is important because any incident scheduled onto an ambulance restricts the availability of that ambulance for incidents later in the sequence. The hybrid TS+CH approach improves the FCFS approach by using TS to vary the sequence of incidents followed by a CH to generate a new feasible solution. Different sequences are investigated through single pairwise swaps of individual incidents. Tabu Search methodology investigates the neighbourhood of solutions obtained by making a single swap from an incumbent solution, accepting the swap resulting in the best solution from that neighbourhood, and then investigating the new neighbourhood around the updated incumbent solution obtained with the accepted swap. Long term memory stores the global best solution and a tabu list of the accepted swaps which are prohibited in later iterations to prohibit cycling of incumbent solutions. The TS algorithm, applied for each horizon in the dynamic model, is shown in Figure 6-7. This process diagram shows the hybrid heuristic continuing to search neighbourhoods for improving solutions until a stopping condition is met. The search is initialised with a solution from the CH as an incumbent solution. Neighbouring solutions are then explored until either the entire neighbourhood has been investigated or a pre-selected number of solutions searched. If any of the searched solutions improve the current global solution, they will replace the global solution. The best solution found from the neighbourhood is then selected as the new incumbent solution for the next neighbourhood and the transition from the old incumbent to the new incumbent placed on the tabu list to prevent cycling between solutions. When the size of the tabu list exceeds a predefined limit, the oldest entry will be deleted.

Neighbourhoods become quite large when there are a large number of incidents with $size_{NBH} = \sum_{n=1}^{I_{max}-1}(n) - size_{tabu\_list}$. For this reason, a limit is placed on the number of incident swaps that will be explored from each neighbourhood. The entire neighbourhood is explored if $size_{NBH}$ is smaller than the limit. Otherwise, swaps are selected in order of anticipated benefit so that good swaps are more likely to be sampled. This smart swap method is based on overtime, delay time, tardy time and makespan from the incumbent solution.

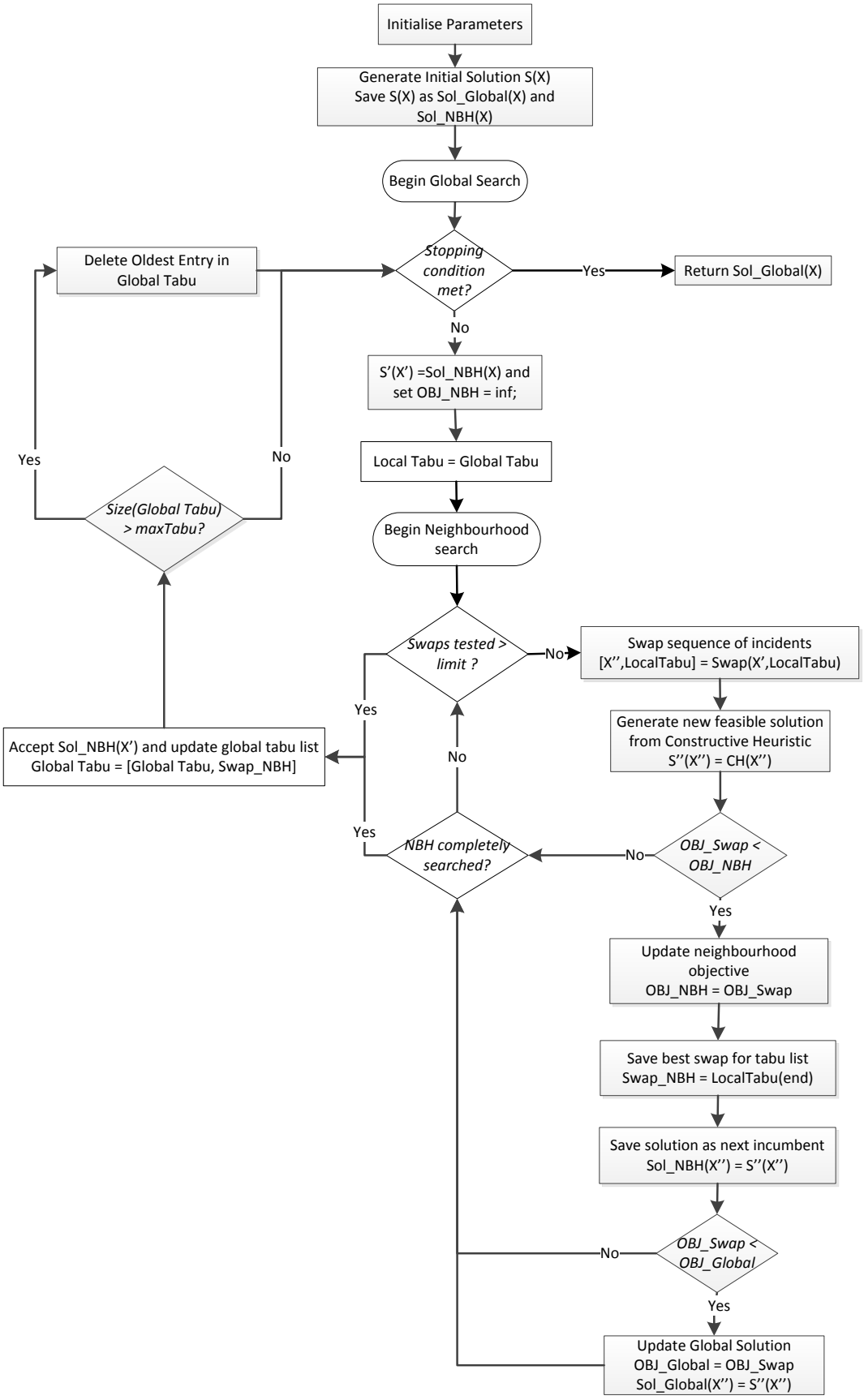


Figure 6-7 Process diagram for the hybrid TS+CH solution approach to the static model

6.2.3.1 Smart Swap Method

The smart swap method determines the order in which pairwise swaps are tested, which is important when only a section of the neighbourhood is searched. This is expected to be an improvement over selecting random swaps as it uses available information to identify incidents with poor performance measures resulting from their assignment in the incumbent solution. It is assumed that these incidents can benefit from fewer restrictions on assignment and so make good candidates to place earlier in the incident sequence.

Each incident is assigned a sequencing rating, *SQ(i)*, to indicate the expected benefit from raising the position of incident *i* in the incident sequence. Incidents that are tardy, have a long makespan, contribute heavily to overtime or suffer from dispatch dates much later than their release date will have a higher rating than those that do not. All terms in the equation for *SQ(i)* are measured in units of time (minutes) and are balanced by a set of weights, $\overline{\boldsymbol{W}}$. The equation to determine ratings is constructed such that it will always be positive and can be written as $SQ(i) = \overline{\boldsymbol{W_1}}(\max(0, r_{ia} - T_i)) + \overline{\boldsymbol{W_2}}(\max(0, d_{ia} - R_i)) + \overline{\boldsymbol{W_3}}\left(\max\left(0, c_{ia} - E_f * x_{iaf}\right)\right) + \overline{\boldsymbol{W_4}}(c_{ia} - d_{ia})$.

Experimental tests of the TS+CH heuristic showed that equal weights produced good solutions. A variation of *SQ(i)* using inverted makespan to prioritise Shortest Processing Time (SPT) rather than Longest Processing Time (LPT) was also tested, as SPT is known to be effective for some scheduling formulations. For the integrated shift and ambulance scheduling problem, use of LPT in the smart swap methodology actually returned better solutions. This is likely due to time windows and possibly resource constraints. Stringent hospital requirements can lead to long processing times for incidents and so LPT, in some cases, will indicate incidents with less flexible assignment options. This provides more incentive to schedule the incident early than a shorter processing time. Due to time windows, incidents with longer processing times are more likely to be unsuitable to schedule onto ambulances where other incidents are already scheduled because there may exist insufficient time to process the incident within the appropriate time window. Further work on the smart swap algorithm could investigate different measures in the smart swap sequencing rating. Alternative approaches could be measuring amount of slack

between arrival times and due date, rather than tardiness, or directly assigning penalties for incidents where fewer hospital choices exist.

After sequencing ratings for each incident have been determined, two incidents must be selected for the pairwise swap. The incident with the highest rating is selected as the incident that will be positioned earlier in the incident sequence ($\boldsymbol{I}_{\uparrow}$). Options for the secondary incident to be swapped into a later position ($\boldsymbol{I}_{\downarrow}$) must appear earlier in the incumbent sequence than $\boldsymbol{I}_{\uparrow}$. From the possible options, the incident with the lowest sequencing rating will be selected. A pairwise swap is then made such that *Position* ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$) = *Position* ($\boldsymbol{I}_{\downarrow}$,$\boldsymbol{I}_{\uparrow}$) and a new solution found from the CH.

Certain swaps are prohibited. These include previously visited swaps which are prevented by the tabu list in long term memory. Additionally, short term memory stores all of the pairwise swaps tested for the current neighbourhood to ensure that each pairwise swap returns an untested sequence. Finally, if $\boldsymbol{I}_{\uparrow}$ is already the first incident in the sequence, or all possible options for $\boldsymbol{I}_{\downarrow}$ are prohibited, then the incident with the next highest sequencing rating will be selected until an allowable swap is found. An example illustrating this process is shown in Figure 6-8.

Previous Incumbent					
Global Tabu List =	(P2,P3), (P1,P5), (P4,P5), (P1,P2)				
OBJ_{NBH} =	∞				
Position	P1	P2	P3	P4	P5
Incident i	I3	I5	I2	I1	I4
Sequencing Rating *SQ(i)*	3	1	5	4	2
Swap 1 of 5					
Local Tabu List:	∅		$OBJ_{NBH} = \infty$		
$\boldsymbol{I}_{\uparrow}$	I2	P3	*highest score*		
$\boldsymbol{I}_{\downarrow}$	[I5, I3]	[P2, P1]	*in order of lowest score*		
Tabu options		(P2,P3)			
Best ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$)	(I2,I5)	(P3,P1)			
OBJ_{swap} =	*7*				
Swap 2 of 5					
Local Tabu List:	(P1,P3)		OBJ_{NBH} =*7*		
$\boldsymbol{I}_{\uparrow}$	I2	P3	*highest score*		
$\boldsymbol{I}_{\downarrow}$	[I5, I3]	[P2, P1]	*in order of lowest score*		
Tabu options		(P2,P3), (P1,P3)			
Best ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$)	∅				
$\boldsymbol{I}_{\uparrow}$	I1	P4	*2nd highest score*		
$\boldsymbol{I}_{\downarrow}$	[I5, I3,I2]	[P2, P1,P3]	*in order of lowest score*		
Tabu options		(P4,P5)			
Best ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$)	(I1,I5)	(P4, P2)			
OBJ_{swap} =	*6*				
Swap 3 of 5					
Local Tabu List:	(P1,P3), (P2,P4)		OBJ_{NBH} = *6*		
$\boldsymbol{I}_{\uparrow}$	I1	P4	*2nd highest score*		
$\boldsymbol{I}_{\downarrow}$	[I5, I3,I2]	[P2, P1,P3]	*in order of lowest score*		
Tabu options		(P4,P5), (P2,P4)			
Best ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$)	(I1,I3)	(P1,P2)			
OBJ_{swap} =	*7*				
Swap 4 of 5					
Local Tabu List:	(P1,P3), (P2,P4), (P1,P4)		OBJ_{NBH} = *6*		
$\boldsymbol{I}_{\uparrow}$	I1	P4	*2nd highest score*		
$\boldsymbol{I}_{\downarrow}$	[I5, I3,I2]	[P2, P1,P3]	*in order of lowest score*		
Tabu options		(P4,P5), (P2,P4), (P1,P4)			
Best ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$)	(I1,I2)	(P4,P3)			
OBJ_{swap} =	*8*				
Swap 5 of 5					
Local Tabu List:	(P1,P3), (P2,P4), (P1,P4),(P3,P4)		OBJ_{NBH} = *6*		
$\boldsymbol{I}_{\uparrow}$	I1	P4	*2nd highest score*		
$\boldsymbol{I}_{\downarrow}$	[I5, I3,I2]	[P2, P1,P3]	*in order of lowest score*		
Tabu options		(P4,P5), (P2,P4), (P1,P4),(P3,P4)			
Best ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$)	∅				
$\boldsymbol{I}_{\uparrow}$	I3	P1	*3rd highest score*		
$\boldsymbol{I}_{\downarrow}$	∅		*in order of lowest score*		
$\boldsymbol{I}_{\uparrow}$	I4	P5	*4th highest score*		
$\boldsymbol{I}_{\downarrow}$	[I5,I3,I1,I2]	[P2,P1,P4,P3]	*in order of lowest score*		
Tabu options		(P1,P5),(P4,P5)			
Best ($\boldsymbol{I}_{\uparrow}$,$\boldsymbol{I}_{\downarrow}$)	(I1,I5)	(P2,P5)			
OBJ_{swap} =	*9*				
New Incumbent					
Global Tabu List =	(P2,P3), (P1,P5), (P4,P5), (P1,P2), (P1,P3)				
OBJ_{NBH} =	6				
Position	P1	P2	P3	P4	P5
Incident i	I2	I5	I3	I1	I4

Figure 6-8 Example showing selection of pairwise swaps within a single neighbourhood using the smart swap method

6.2.3.2 Constructive Heuristic for the Hybrid TS+CH

This solution rebuilding algorithm is based on the CH presented above. It is modified to include a probability that an incident will be allowed to be tardy that is based on the number of incidents of the same priority type that are already tardy and the remaining number of incidents still to be assigned. Tardy responses are avoided, where possible, in any instance requiring a new ambulance crew to be introduced into the system. In this instance, three stochastic parameters are introduced to allow variation of the ambulance type, ambulance station and initial shift selected for the new ambulance crew. In the previous CH, the cheapest ambulance type, the closest ambulance station and the earliest possible shift were always selected. The new CH selects ambulance type, ambulance station and assigned shift randomly with probabilities established respectively by the ambulance type required, travel time from each ambulance station and overtime incurred. This CH is run multiple times and the best solution selected as a solution in the current neighbourhood. If the overall solution improves, it replaces the solution in long term memory. The stopping condition is 200 iterations without an improved solution or a CPU time of 1000 seconds, whichever occurs first.

The algorithms for the hybrid TS+CH; the modified CH, used to rebuild solutions in the hybrid TS+CH; and the new sub-function for assigning new ambulances, called by the modified CH, are shown in Figures 6-8, 6-9 and 6-10 respectively.

TS+CH hybrid heuristic

Generate initial incumbent solution x from CH algorithm
Store $x \rightarrow x^{*}$, $f(x) \rightarrow f^{*}(x)$, $x \rightarrow x^{\dagger}$, $f(x) \rightarrow f^{\dagger}(x)$, $TL \rightarrow \emptyset$, $i = 0$
while solve time < time limit && i < limit for iterations without improvement
    Set $x = x^{\dagger}$, $f(x) = f^{\dagger}(x)$, Set $x^{\dagger\dagger} = x^{\dagger}$, $f^{\dagger\dagger}(x) = \infty$
    while count < iteration limit for inner loop
        $x' = x$
        Calculate measure of benefit of swapping each incident j: $u(j) = g(delay, tardy, overtime, makespan)$,
        $j_1 = \arg\left(\max_{j \in \tilde{\mathbf{j}}} u(j)\right)$ & $j_2 = \arg\left(\min_{j \in \tilde{\mathbf{j}}\ s.t.\ j < j_1} u(j)\right)$
        Swap $x'(j_1) \leftrightarrow x'(j_2)$ and add $(j_1, j_2) \rightarrow TL$
        if $size(TL)$ > tabu list limit
            Remove oldest entry
        end if
        $f'(x') = Rebuild\_CH(x')$ [rebuild solution with latest sequence]
        if f'(x) < $f^{\dagger\dagger}(x)$
            $x^{\dagger\dagger} = x'$, $f^{\dagger\dagger}(x) = f'(x)$ *[update neighbourhood solution]*
        end if
    end while
    Set $x^{\dagger} = x^{\dagger\dagger}$, $f^{\dagger}(x) = f^{\dagger\dagger}(x)$
    if $f^{\dagger}(x) < f(x)$
        $x = x^{\dagger}$, $f(x) = f^{\dagger}(x)$ *[update global solution]*
    end if
end while

Figure 6-9 Algorithm for the TS+CH hybrid heuristic to solve the static model

CH for Hybrid TS+CH

For *i= 1 to* I_{max}

$AllowTardy = \begin{cases} 1, & rand < TarRej(P_{ip}, Q_p) \\ 0, & otherwise \end{cases}$

Select $\boldsymbol{F'} \subset \boldsymbol{F}$ s.t. $B_{\boldsymbol{F'}} \leq U_j$ && $E_{\boldsymbol{F'}} \geq R_j$

Select $\boldsymbol{A'} \subset \boldsymbol{A}$ s.t. $\xi_{i\boldsymbol{A'}} = 1$ && $\left(\sum_{f \in \boldsymbol{F}} \boldsymbol{z_{A'sf}}\right) \times D_i \geq R_i + \theta_{is}$

Select $\boldsymbol{H'} \subset \boldsymbol{H}$ s.t. $\gamma_{\boldsymbol{iH'}} = 1$

Set *assigned = 0*

While *assigned = 0*

If *isempty(**A'**)*

Run Assign_New2

assigned = 1

ElseIf $\min_{a \in \boldsymbol{A'}} \left(\max_{\mathrm{f} \in \mathbf{F}} \sum_{s \in \boldsymbol{S}} \left(z_{\boldsymbol{A'}sf} \times \theta_{is}\right) > D_i - R_i\right)$

Run Assign_New2

assigned = 1

Else

$a = \arg\left(\min_{\mathrm{a} \in \mathbf{A'}} \left(\max_{\mathrm{f} \in \mathbf{F}} \sum_{s \in \boldsymbol{S}} \left(z_{\boldsymbol{A'}sf} \times \theta_{is}\right)\right)\right)$

$s = \arg\left(\max_{s \in \boldsymbol{S}} \left(\max_{\mathrm{f} \in \mathbf{F}} z_{asf}\right)\right)$

If $\left(\sum_{f \in \boldsymbol{F'}} z_{asf}\right) > 0$

$f = \arg\left(\min_{f \in \boldsymbol{F'}} \left(B_f + M\left(1 - z_{asf}\right)\right)\right)$

Else If $\left(\sum_{f \in \boldsymbol{F}} z_{asf}\right) > 0$

$\boldsymbol{F_{block}} = \{f-2, f-1, f, f+1, f+2\}$
$\forall f \in \boldsymbol{F}$ s.t. $z_{asf} = 1$

$\boldsymbol{F''} = \boldsymbol{F'} \setminus \boldsymbol{F_{block}}$

If *isempty(**F''**)*

f *= 0*

$\boldsymbol{A'} = \boldsymbol{A'} \setminus a$

Else f *=* $\boldsymbol{F''}$*(1)*

End If

End If

If $f > 0$

$d = \max(R_i, B_f)$

$\mathrm{r} = \mathrm{d} + \theta_{is}$

$c = r + \min_{\mathrm{h} \in \mathbf{H}} (\psi_{ih} + \zeta_{sh})$

If $r > U_i$

$\boldsymbol{A'} = \boldsymbol{A'} \setminus a$

ElseIf $r > D_i - M \times AllowTardy$

$\boldsymbol{A'} = \boldsymbol{A'} \setminus a$

Else *(d,r,c)* = DisjunctionCheck*(d,r,c,i,a,f,s,h)*

If $r > U_i$

$\boldsymbol{A'} = \boldsymbol{A'} \setminus a$

ElseIf $r > D_i - M \times AllowTardy$

$\boldsymbol{A'} = \boldsymbol{A'} \setminus a$

ElseIf $r > E_f$

$\boldsymbol{A'} = \boldsymbol{A'} \setminus a$

CH for Hybrid TS+CH cont'd

Else Save selected path
$x_{iaf} = 1, y_{ish} = 1, z_{asf} = 1$
$d_{ia} = d, r_{ia} = r, c_{ia} = c$
$n_i = 0, o_{af} = \max(o_{af}, c_{ia} - E_f)$,
$assigned = 1$
End If
End If
End If
End If
End While
End For
Return $\sum_{k \in K}(\omega_k \sum_{a \in A_k} \sum_{s \in S} \sum_{f \in F} z_{asf} + \sigma_k \sum_{a \in A_k} \sum_{f \in F} o_{af})$

Figure 6-10 Algorithm for the constructive part of the TS+CH heuristic for the static model

Assign New 2

Load stochastic parameters
Select k from $(k \mid i)$, depending on type of ambulance requested by incident i
Add new ambulance a of type k ($\boldsymbol{A_k} = \{\boldsymbol{A_k}, a\}$)
Assign earliest dispatch time (i.e. $d_{ia} = R_i$)
Find earliest shift options $f = \arg(\max_{f \in \boldsymbol{F}'}(B_f))$ where $\boldsymbol{F}' \subset \boldsymbol{F}\ s.t\ B_{\boldsymbol{F}'} < R_i$
Select hospital $h = \arg(\min_{\mathrm{h} \in \boldsymbol{H}'}(\psi_{ih} + \zeta_{sh}))$ $w$here $\boldsymbol{H}' \in \boldsymbol{H}$ s. t. $\gamma_{i\boldsymbol{H}'} > 0$
Identify completion location $c_l = \begin{cases} L_h, & \gamma_{i\boldsymbol{H}'} > 0 \\ L_i, & otherwise \end{cases}$
Set probability for each ambulance station $s \in \boldsymbol{S}$

$$P(s) = \begin{cases} \exp\left(\frac{-\theta_{is}}{U_i - R_i}\right), & \theta_{is} \le D_i - R_i \\ 0, & \theta_{is} > D_i - R_i \end{cases}$$

$$\bar{P}(s) = \frac{P(s)}{\sum_{s' \in \boldsymbol{S}} P(s')}$$

If $\sum_{s \in \boldsymbol{S}} \bar{P}(s) = 0$

Then select ambulance station with shortest response time
$s = \arg\left(\min_{s' \in \boldsymbol{S}}(\theta_{is'})\right)$

Else select random s from $\bar{P}(s)$
End If
Determine expected $c_{ia}^{'}$ & $o_{af}^{'}$
If $o_{af}^{'} > 0$ && $B_{f+1} + \min_{s \in \boldsymbol{S}} \theta_{is} \le \min(U_i, D_i + M \times AllowTardy)$
(i.e. incident i can be delayed until the next shift $f+1$
If $rand \le OverAccept$
Delay incident i and update variables
$s = \arg(\min_{s' \in \boldsymbol{S}}(\theta_{is'}))$, $f = f + 1, d_{ia} = B_f, o_{af} = 0\ etc.$
End If
End If

Figure 6-11 Updated algorithm for assigning new ambulances in the static model

6.2.4 MIP solver (CPLEX)

The model was formulated as a MIP model and solved using OPL for IBM ILOG CPLEX Optimization Studio version 12.4 run on a DELL laptop using an Intel Core i7 processor with 8GB of physical memory. The data required as input included the maximum number of ambulances allowed of each type in addition to the full set of parameters for each job and expected processing times. To determine a good initial number of ambulances, the output was taken from the CH presented above and developed for use as input for the MIP model. This has the benefit of reducing the initial number of ambulances available to a reasonable guess, which reduces the number of variables in the problem for a faster solution. However, reducing the number of available ambulances risked excluding potentially optimal solutions. One additional ambulance of each type was added to the CH solution to give the MIP more options without greatly inflating the number of variables.

6.3 RESULTS AND DISCUSSION

In this section, small scenarios are used to test each solution model to check quality and solve times. The TS+CH heuristic, found to be the best performing heuristic for large scenarios, is then used to solve the model to create a shift schedule for one week.

6.3.1 Small Problem Size

Several small scenarios, having increasing numbers of incidents from Data Set A, are run 10 times. Table 6-1 shows the number of incidents in these scenarios with total time elapsed between the first incident arrival and the last incident arrival. This indicates the amount of time in the real world which each scenario covers.

Table 6-1 Total time elapsed between arrival of first incident and arrival of 'n^{th}' incident.

Number of Incidents	10	20	30	40	50	60	70	80	90	100
Total Time Elapsed (mins)	104	153	199	240	306	335	360	380	420	454

The MIP solver was run with a stopping condition of 12 hours (43,200 seconds) of CPU time or optimality, whichever occurred first. The CH+TS algorithm had dual stopping conditions of 200 iterations without any improved solution found or a CPU time limit of 1000 seconds, whichever occurred first. The results are shown in Table 6-2. The units for the objective function value are Weighted Ambulance

Hours (WAH). This reflects the cost of running ambulance services by the number of hours worked by ambulances of all types. Each hour of a regular shift for the cheapest ambulance adds 0.1 to the objective function value, so that a full shift of 10 hours is equivalent to 1 WAH while each minute of overtime contributes $1/30^{th}$ of the cost of an hour of regular time on the same ambulance. More expensive ambulances are weighted to contribute more WAH for the same amount of time spent working.

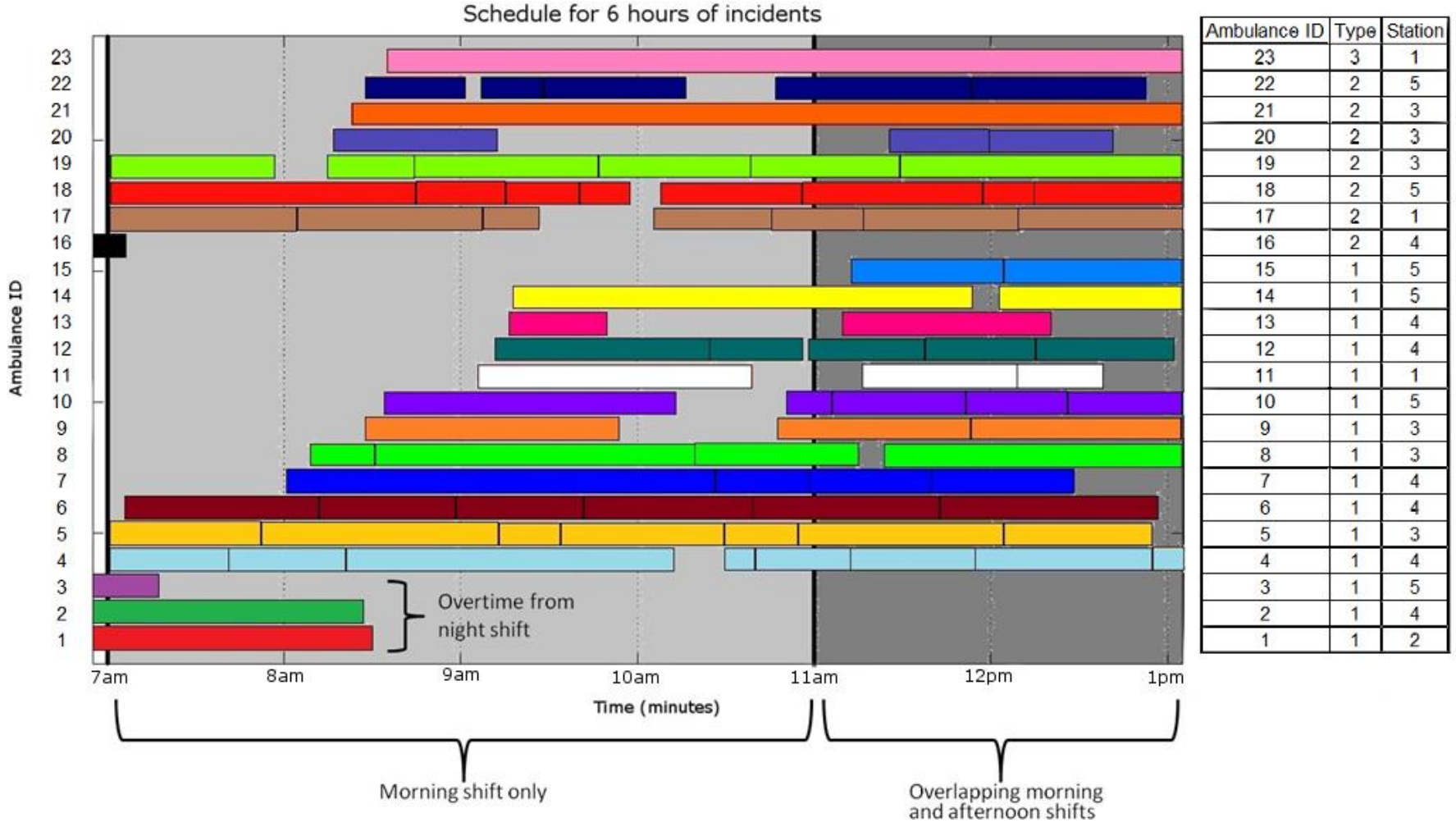
Results from the CH algorithm were obtainable within seconds; however, the CH solutions were outperformed by solutions from the TS+CH hybrid heuristic and MIP solver. Feasible solution to the MIP model, found using an MIP solver with a time limit of 12 hours of CPU time, were able to be solved for problem sizes of up to 50 incidents (approximately 5 hours of real time). Exact solutions, within this time limit, were only found for very small size problems (scenarios with < 15 incidents). Larger problems, for example 50 incidents, were able to find feasible solutions within this time but did not find optimal solutions even with relaxing the time limit constraint to allow the model to run for several days. The solution software began to fail due to the amount of memory required for the MIP model. This limited the usefulness of the MIP solver, making it unsuitable for creating strategic shift schedules covering multiple shifts. For this reason, it was chosen to present the results from the MIP solver with the time limit of 43 200 CPU seconds (12 hours of CPU time) to have feasible results for as many scenarios as possible with a reasonable amount of time for comparison against meta-heuristics.The TS+CH hybrid heuristic is able to match the MIP solution for the smallest problem, and is faster than the MIP for all but the smallest problem. While the heuristic is not guaranteed to converge to optimal solutions, the average solution begins to outperform the time limited MIP solver when there are 35 or more incidents, and was able to solve larger problems that the MIP solver could not. It is also noted that the average solution at 70 incidents is non-monotonic. This is an effect of similar initial feasible solutions and variability in non-optimal solutions.

A segment of the resulting best schedule for 90 incidents is shown in Figure 6-12. This allows investigation of the usefulness of schedules obtained from the model. It shows which ambulance crews were assigned to each shift, which station was assigned as their home station and how much time was spent dealing with

incidents. It indicates the overtime that was unavoidable at the end of the night shift from incidents that continued past the end of the shift, and a number of incidents where it was acceptable to delay a start until fresh ambulances came on shift. Incidents which utilise ambulances for excessive periods of time are also easily identified. There are two incidents in the time period covered by the schedule that are in service for > 6 hours. In both cases, this is due to excessive ramping time. The schedule presented also reveals a potential flaw in the scheduling process. Schedules rebuilt from the hybrid heuristic only assign new ambulances when no previously assigned ambulance is available. This creates schedules where some new ambulances may be waiting for two hours before dealing with their first incident on a shift and workloads are unbalanced.

Table 6-2 Results from each solution approach for the static model

Number of Incidents	**CH**		**MIP**		**Hybrid TS+CH**		
	CPU Time (secs)	*Objective Function Value (WAH)*	*CPU Time (secs)*	*Objective Function Value (WAH)*	*CPU Time (secs)*	*Average Objective Function Value (WAH)*	*Best Objective Function Value (WAH)*
5	0.03	7.47	1.65	7.43	39.08	7.43	7.43
10	0.04	14.21	474.41	12.46	86.92	12.56	12.49
15	0.05	19.71	*time limit reached*	15.40	316.43	16.82	16.57
20	0.09	24.21	*time limit reached*	17.20	377.26	18.80	18.23
25	0.11	30.21	*time limit reached*	20.06	717.45	22.18	20.82
30	0.12	33.21	*time limit reached*	22.38	686.12	24.72	22.66
35	0.14	35.21	*time limit reached*	25.93	1000.57	27.08	26.10
40	0.18	40.71	*time limit reached*	29.92	1000.93	32.82	30.97
45	0.20	44.71	*time limit reached*	38.90	1000.06	35.02	33.10
50	0.21	44.71	*time limit reached*	39.90	1000.17	35.76	33.83
55	0.23	44.71	*time limit reached*	-	1000.45	36.23	34.25
60	0.25	44.71	*time limit reached*	-	1000.92	36.27	34.29
65	0.29	44.71	*time limit reached*	-	1000.68	37.54	36.26
70	0.29	44.71	*time limit reached*	-	1001.84	37.43	36.63
75	0.37	46.71	*time limit reached*	-	1000.29	39.37	37.06
80	0.41	48.21	*time limit reached*	-	1000.68	40.66	38.97
85	0.43	48.21	*time limit reached*	-	1002.07	41.44	40.61
90	0.45	49.71	*time limit reached*	-	1001.12	44.12	42.15
95	0.46	51.21	*time limit reached*	-	1000.72	46.23	43.74
100	0.50	53.21	*time limit reached*	-	1002.38	47.31	45.24



Ambulance ID	Type	Station
23	3	1
22	2	5
21	2	3
20	2	3
19	2	3
18	2	5
17	2	1
16	2	4
15	1	5
14	1	5
13	1	4
12	1	4
11	1	1
10	1	5
9	1	3
8	1	3
7	1	4
6	1	4
5	1	3
4	1	4
3	1	5
2	1	4
1	1	2

Figure 6-12 Schedule from hybrid heuristic for ambulances responding to incidents across 6 hours.

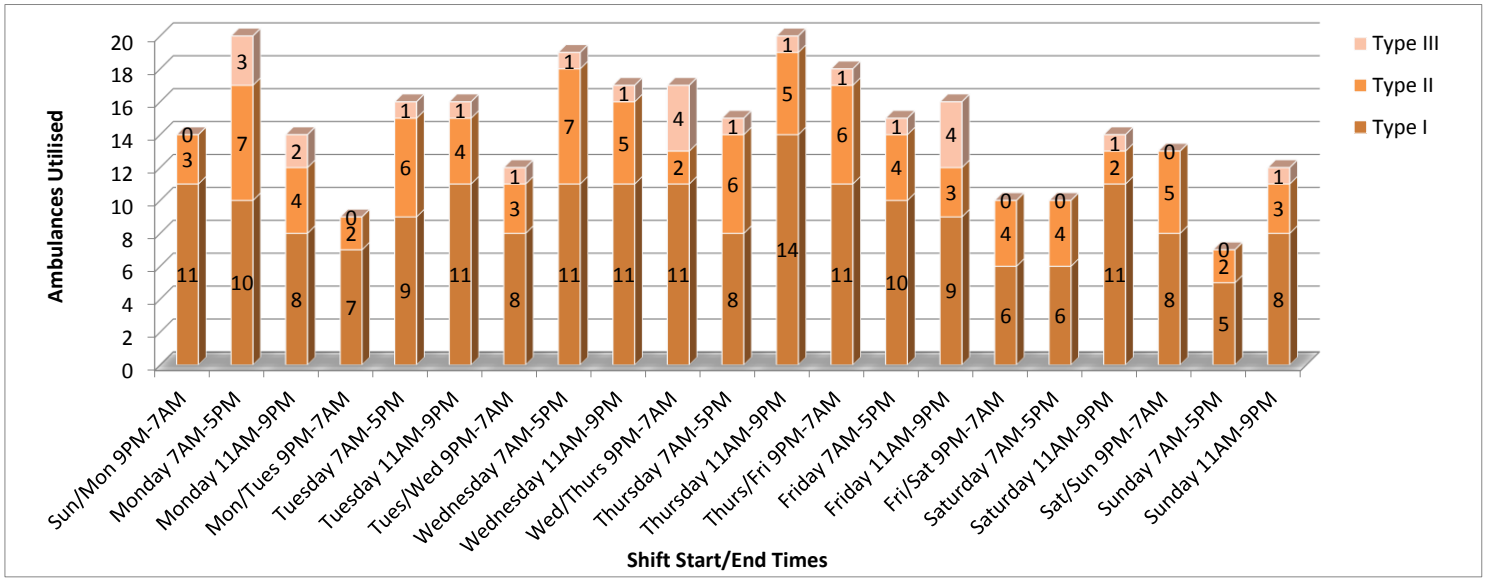


Figure 6-13 Number of ambulance crews, of each type scheduled to work each shift in weekly output from the static model

6.3.2 Weekly schedule

The hybrid heuristic was also used to solve the problem for one week's worth of incidents. The results are shown in Table 6-3. The best solution from the hybrid heuristic is able to improve the CH solution by reducing the number of Type II ambulances and increasing the number of Type III ambulances. However, the results utilise more ambulances per shift for the five ambulance stations investigated than were present in workforce utilisation data (7.9 ambulances per shift). This was expected as the static model introduces additional work through forcing each ambulance to return to its home station after each incident. It is also possible that compelling solutions to always meet certain performance requirements and simplification of shift patterns inflated the number of required ambulances. Further work is required to develop this formulation into a dynamic formulation.

Response and dispatch to clear times from the schedule with the lowest objective function for the cost of running ambulance services are shown in Table 6-4. Through adding additional ambulances where necessary, the static model is able to produce excellent response times. All emergency incidents are met within 10 mins and the 50^{th} and 90^{th} percentile response times, of 5.6 and 9.6 minutes respectively, are improvements on the targets of 8 minutes and 16 minutes. For all incidents, a 25% improvement in the median response time is observed when compared with the percentile response time shown in Table 5-5 from actual events.

Table 6-3 Results for one week of incidents from the static model

	CH			Hybrid TS+CH		
Number of Incidents	*CPU Time (secs)*	*Objective Function Value (WAH)*		*CPU Time (secs)*	*Objective Function Value (WAH)*	
					Average	*Best*
1341	25.63	631.56		1027.71	585.68	577.70
Ambulance Type	*Total number of ambulances*	*Average number of ambulances per shift*	*Average overtime per shift (mins)*	*Total number of ambulances*	*Average number of ambulances per shift*	*Average overtime per shift (mins)*
Type I	34	8.90	170.69	37	8.86	196.25
Type II	21	5.67	142.72	15	4.00	90.85
Type III	2	0.67	4.43	6	1.14	3.33
ALL	57	15.24	317.84	58	14.00	290.43

Table 6-4 Performance of the best schedule found with the static model

Priority	**Average Response Time (mins)**	**Percentile response time (mins)**		**Percentage met in**		
		50^{th}	*90^{th}*	*< 10 mins*	*< 30 mins*	*< 60 mins*
ALL	17.08	7.86	41.86	61.98%	81.29%	95.42%
Emergency	6.09	5.58	9.6	93.23%	100%	100%
Urgent	15.07	8.5	37	57.48%	85.82%	100%
Non Urgent	27.81	18.06	65.51	39.77%	62.20%	87.96%

The number of ambulances utilised each shift is illustrated in Figure 6-13. This shows the number of ambulances each shift and the breakdown in vehicle type. It is observed that:

- Weekday shifts require more ambulances than weekend shifts to meet higher demand;
- Day shifts during the week also require more ambulances than overnight shifts during the week as, even though day shifts have a period of overlap, they must still deal with peak demand;
- Weekend shifts generally require fewer Type III ambulances;
- Type I is most commonly required vehicle type;
- Type III not required every shift.

Station allocation for each ambulance during the week is also investigated. The number of shifts for ambulance crews based at each station is shown in Table 6-5. This also compares results of the number of ambulance hours against the real data. Comparisons between the static model and the real ambulance workforce data should be interpreted with caution, however, as the static model used a case study based on ambulance demand level five years later than the workforce data available. The results in Table 6-5 show that the static model schedules approximately double the number of ambulances required compared to the number of ambulances in the workforce data. Not all of this can be attributed to increasing demand between the time at which workforce data is available and the more recent data used to create the case study. The static model overestimates the number of ambulances. There are two reasons most likely to be the cause of this overestimation. Firstly, the requirement to return to the correct ambulance base prior to being dispatched to another incident decreases the amount of time for which an ambulance is available. This can be addressed by formulating a dynamic model. Secondly, the shift schedules were limited to three options per day and forced ambulances to start each shift at the same ambulance station. This is expected to create a less efficient schedule than a more flexible approach. However, increasing the number of shift options increases the size of the problem. The hybrid heuristic developed in this chapter is expected to be able to solve the larger problem but more time will be required.

Table 6-5 Comparison of ambulance hours scheduled at each ambulance station from the static model and real data

	1: Northgate	2: Kedron Park	3: Chermside	4: Spring Hill	5: Roma Street
Ambulance Shifts *(static model, 1 week)*	52	45	78	70	51
Ambulance hours *(static model, 1 week)*	520	450	780	700	510
Ambulance hours (2006/07 workforce data)	273	203	413	431	291

6.3.3 Sensitivity Analysis

This section further explores the capability of the hybrid TS+CH heuristic by analysing results for select problem sizes. The purpose of this analysis is to explore the performance of the new heuristic and the sensitivity of the weighting parameters used in the objective function in order to determine the significance of including overtime within the objective. Problems containing 40 incidents and 90 incidents are selected. These scenarios have a real world timespan of 4 hours and 7 hours respectively. The scenario with 40 incidents is small enough so that all incidents receive a response on a single shift while the scenario with 90 incidents requires two shifts to be scheduled. These problems also map to solutions where the TS+CH outperformed the MIP solver.

Figure 6-14 shows solutions from the hybrid heuristic at successive iterations. In both scenarios, the most significant improvements were found in the first neighbourhood with fewer solutions were found in later neighbourhoods. As the most valuable improvements to the solutions occur near the beginning of the solution period, time and/or number of iterations are reasonable stopping conditions for obtaining good solutions from the hybrid heuristic. Moving average data, plotted against both iterations and solution time (Figure 6-15), supports the idea that improving solutions can be found quickly with smaller improvements happening over a longer period time. A stopping condition of a 1000 second time limit is adequate.

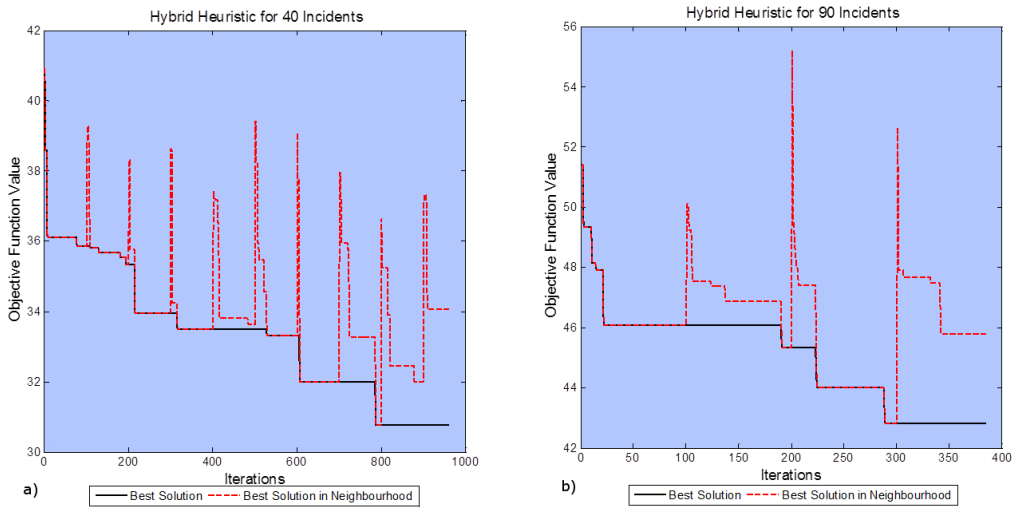


Figure 6-14 Solution values from the hybrid TS+CH heuristic for the static model

Varying the objective weights tests the sensitivity of the model with respect to relative ambulance type costs (by testing a steeper cost gradient), and sensitivity to the inclusion of overtime. The modified weight sets tested are shown in Table 6-6. The solutions from the TS+CH heuristic from these weight sets are shown in Table 6-7. The objective function value is given for the weights tested (denoted as OFV) and the equivalent value using standard weights (OFV*) in Weighted Ambulance Hours. The best and worst solutions are compared with the 10^{th}, 50^{th} and 90^{th} percentiles. For the standard weights, these results indicate a potential skew away from the best solutions found.

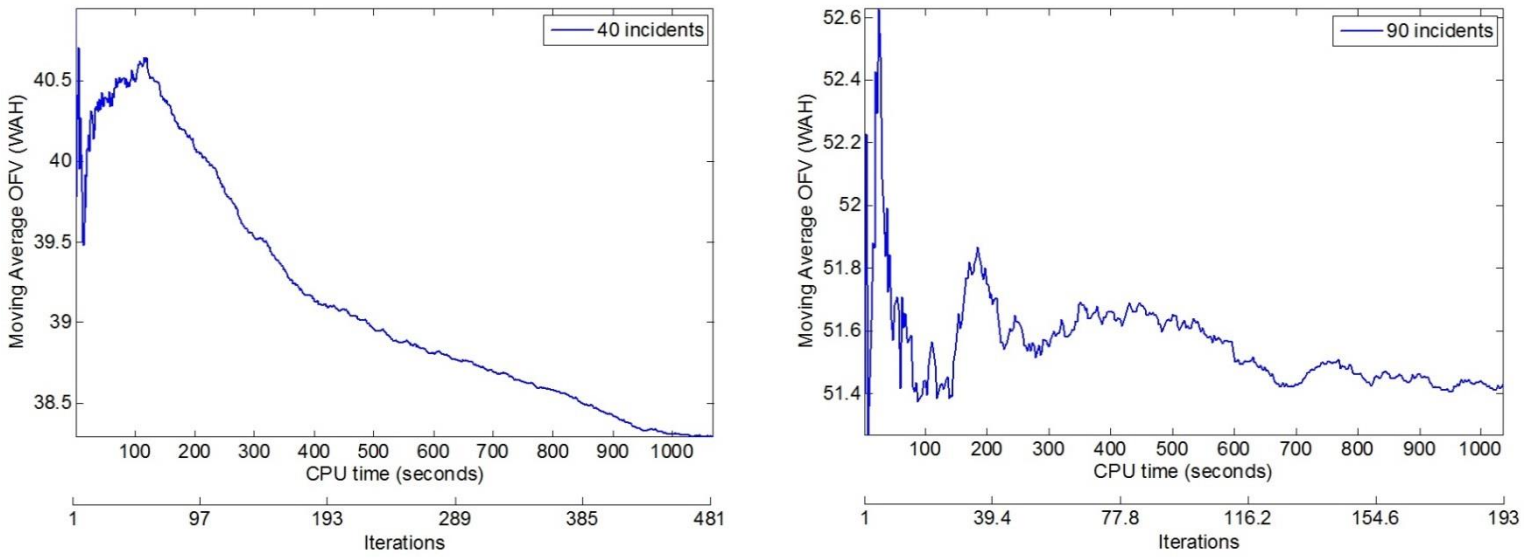


Figure 6-15 Moving average of the objective function for the static model

Table 6-6 Various weights used in the objective function

Weights Set	Weighted cost of ambulance of type k for one shift (ω_k)			Cost of one unit of overtime for an ambulance of type k (σ_k)		
	Type I	Type II	Type III	Type I	Type II	Type III
Standard weights	2	1.5	1	0.006667	0.005000	0.003333
Varied weights	3	2	1	0.01000	0.006667	0.003333
Regular time weights only	2	1.5	1	0	0	0
Overtime weights only	0	0	0	0.2	0.15	0.1

Table 6-7 Comparison of Objective Function Values (measured in WAH) for assorted weights in the static model

Number of Incidents	Percentile Solution	Standard weights	Varied weights		Regular time weights only		Overtime weights only	
		OFV	*OFV*	*OFV**	*OFV*	*OFV**	*OFV*	*OFV**
40	*1st (Best)*	**30.21**	43.91	31.84	29.5	30.73	28.57	40.95
	10th	31.24	44.87	31.81	30.5	31.68	28.90	34.96
	50th (Median)	33.23	47.18	33.46	31.5	32.9	29.94	36.5
	90th	33.93	48.75	33.94	32.5	33.89	3123	36.04
	100th (Worst)	34.29	49.22	34.56	33.5	35	33.33	35.61
90	*1st (Best)*	41.56	59.89	42.33	40	**41.28**	29.29	47.98
	10th	42.74	62.13	43.43	41.5	43.11	29.87	45.5
	50th (Median)	44.45	63.54	44.51	43	44.64	30.63	46.52
	90th	45.56	65.28	46.04	43.5	45.35	33.76	48.13
	100th (Worst)	46.48	66.32	46.11	44	45.19	36.56	43.71

The components of the best solutions (i.e. the number of ambulances and amount of overtime) from the same assortment of objective function weights are detailed in Table 6-8 and Table 6-9 in order to explore how weights affect particular solution components. The varied weight set, which biases the objective more strongly in favour of cheaper ambulances, has the effect of reducing the number of Type I ambulances when compared with solutions from the standard weights, but increases Type II and/or Type III ambulance numbers and overtime. Therefore, the model is sensitive to the weights used to differentiate ambulance types. Minimising overtime alone indicates that a non-zero amount of overtime is to be expected, as 103.84 mins of overtime for Type I ambulances is common to the best solutions for both the 40 incident and 90 incident scenarios. Solutions found where overtime was not considered (i.e. only regular time has non-zero weights in the objective function) are able to outperform solutions found with the standard weights for several scenarios as seen for the best solutions found for 90 incidents. Weights with the emphasis only on regular time focus the objective on the number of ambulances

used. For standard weights, the number of ambulances used is the largest contributor to the objective function value. For the regular time weights, overtime is still considered within the solution methodology itself because the section of the TS+CH heuristic selecting incidents to swap in each neighbourhood is informed by overtime. Future modifications to the heuristic will need either to retain the approach of considering overtime within the neighbourhood search or the objective, but may not require both.

Table 6-8 Components of best solution for 40 Incidents with hybrid TS+CH algorithm in the static model

Weights Set	Number of Ambulances			Overtime (mins)		
	Type I	Type II	Type III	Type I	Type II	Type III
Standard weights	9	6	2	103.8438	103.5632	0
Varied weights	8	7	4	112.4884	117.2591	0
Regular time weights only	10	5	2	120.5043	86.12471	0
Overtime weights only	11	10	3	103.8438	51.99016	0

Table 6-9 Components of best solution for 90 Incidents with hybrid TS+CH algorithm in the static model

Weights Set	Number of Ambulances			Overtime (mins)		
	Type I	Type II	Type III	Type I	Type II	Type III
Standard weights	14	7	2	103.8438	73.38404	0
Varied weights	12	10	2	103.8438	127.9707	0
Regular time weights only	13	8	2	120.5043	96.10596	0
Overtime weights only	15	10	2	103.8438	56.80335	0

6.4 VARIATIONS

During the development of this model, several variations were considered and explored. While these ideas are not included in the final version of the model, they are recorded here for completeness.

Allow dispatch to new incidents within reasonable time after the end of a shift.

The model presented above prevents ambulances being assigned to new incidents past the time at which they are due to end a shift. In reality, there may be cases where an ambulance will be available after the end of its shift and the closest vehicle for high priority incidents. In such events, it makes sense to dispatch an ambulance after the end of its shift. This variation is a generalisation of the static

model relaxing constraints 6.16 and 6.17 and introducing parameter ϕ_{af}. The new parameter appears in constraints 6.26 and 6.27 replacing constraints 6.16 and 6.17.

Parameter

ϕ_{af} Reasonable amount of time after the end of shift *f*, during which ambulance *a* may still be dispatched and accrue overtime.

Constraints

$$\left(d_{ia} - \phi_{af}\right) - E_f \leq M\left(1 - x_{iaf}\right) \qquad \forall\, i \in I, a \in A, f \in F \tag{6.26}$$

$$B_f - \left(d_{ia} - \phi_{af}\right) \leq M\left(1 - x_{iaf}\right) \qquad \forall\, i \in I, a \in A, f \in F \tag{6.27}$$

By setting ϕ_{af} to zero, this variation becomes the static model presented at the beginning of this chapter. This variation is simple to implement but has not been included in the static because the objective is to minimise overtime through ensuring enough ambulances are available at the beginning of each shift. There is also a lack of information about appropriate values for ϕ_{af}.

Upper limits on overtime for each shift.

This variation places further constraints on overtime values, and requires an additional input parameter, π_{af}, in order to prevent overtime exceeding acceptable amounts each shift and a modified boundary constraint for o_{af}.

Parameter

π_{af} Maximum overtime allowed on ambulance *a* during shift *f*.

Constraint

$$0 \leq o_{af} \leq \pi_{af} \qquad \forall\, i \in I, a \in A, f \in F \tag{6.28}$$

This would allow maximum overtime to vary for each individual ambulance. This is closer to real situations as it allows balancing of overtime workloads to suit ambulance staff. It was elected not to include this parameter in the static model because this should not be a hard constraint, rather it should be represented as a preference; the nature of the parameter should be dynamic, allowing overtime worked on previous shifts to affect the ability to assign overtime later; and minimising overtime is addressed in the objective function. The static model allows

unlimited overtime with the objective of minimising overtime and restrictions on beginning new work after the end of a shift.

Multiple ambulances responding to individual incidents

A static model allowing multiple ambulance responses was briefly explored. While it was rejected due to the additional complexity, the rationale behind the abandoned formulation is discussed here.

The formulation considered incidents were multiple ambulances respond to a single incident. This may be multiple casualties or a single patient requiring additional emergency service staff. For example, additional paramedics may be required to treat or transfer a patient into the back of an ambulance, a specialist paramedic may need to attend the scene, but other, closer units will be dispatched first, to minimise waiting without any medical attention. This proposed variation would also allow the model to directly include ambulances that are specialist response units but are not equipped to transfer patients to hospitals. In the static model presented above, it is assumed that all ambulances can transfer a patient to hospital and the input dictates which ambulances are allowed to respond to which incidents.

Considerations for a model that allow multiple ambulance responses include: relaxing the constraints upon tardy responses for the additional vehicles arriving after the first on scene; linking the stabilisation time for treating patients at the scene of the incident; and ensuring that only one ambulance may be responsible for transferring a patient to hospital. Multiple patients per ambulance may be permitted. These considerations are more suited to a reactive, real time model where information on expected processing times or the severity of an incident (affecting required response times for additional ambulances) would be updated as ambulances arrive at the scene of an incident and paramedics are able to assess the situation. However, additional precedence relationships may still be added into a static model to represent these situations more closely than the assumption that each incident receives exactly one ambulance response.

6.5 IMPLICATIONS AND FURTHER WORK

A new Hybrid TS+CH heuristic has been developed to solve a new, static MIP model for integrated ambulance scheduling and ambulance crew shift scheduling.

Solutions provide schedules that minimise overtime and place an upper bound on the minimum number of ambulance crew shifts required to create a schedule that will satisfy performance requirements. The model can be solved for a small problem containing five ambulance stations co-operating in a metropolitan environment using a standard MIP optimiser; however, the problem quickly becomes intractable for multiple hours of incident arrivals and heuristics are needed to generate strategic shift schedules. The hybrid heuristic is able to provide good solutions within minutes. The results also show that the inclusion of overtime in the objective function value allows overtime to be considered without noticeable adverse effects on minimising the total number of ambulances required.

The practical application is a strategic planning tool for ambulance scheduling. The methodology also provides a basis for formulating ambulance problems using flexible flow shop scheduling and demonstrates hybrid heuristics that are suitable candidates for solving the problem. Further development of this methodology is being undertaken to extend the model to integrate additional shift scheduling rules and dynamic relocation of ambulances.

The schedules produced by the static model also suffered from unbalanced workloads. Additional shift scheduling rules introduced into a new model may go some way to correcting this flaw. The most important improvements to the model can be made through relaxing the static nature of the model. The static model was limited by the assumption, necessary to simplify overtime, that ambulances could only be dispatched to incidents when they were clear and at their base station. This allowed an estimate of the upper bound of required resources but the actual number of required resources is expected to be lower. Further work is needed in the form of dynamic and real time models to more accurately represent the real life situation.

Further work on the static model should extend the case study to include more stations and hospitals. Solving the model with additional ambulance stations at locations that do not yet exist in reality may also provide insight into the effect that opening new stations or relocating a station can have on reducing the required number of ambulances.

The next model will consider dynamic relocation of ambulances with deterministic demand. The real time, dynamic model will need to store information

on both location and status of ambulances at time *t* in order to handle reassignment and redeployment. A rolling horizon approach should be able to be implemented along with insertions for demand. Creating subsets of ambulances, as in the model presented by Haghani and Yang (2007), is a technique that will be considered. These models will give a better estimate of the number of resources required each hour of the day and the utilisation of each ambulance. Meta-heuristics, hybrid heuristics and hyper heuristics will be investigated as potential solutions to the NP-hard real time, dynamic model.

Chapter 7: Dynamic Model

This chapter introduces the dynamic ambulance model that extends the static model presented in Chapter 6. Where the previous model required each ambulance to be dispatched from a static location (that is, only one ambulance station for each ambulance), this model allows ambulances to be dispatched from their last known location and allows ambulances to be relocated to different ambulance stations when they are available.

Solution approaches include modifications of the Constructive Heuristic and hybrid Tabu Search and Constructive Heuristic used for the static model as well as a new Ant Colony Optimisation heuristic and a hybrid Ant Colony Optimisation and Constructive Heuristic. The hybrid TS+CH and ACO+CH heuristics are the most promising for solving this model. The CH and hybrid heuristics are used to solve the model with a rolling horizon approach, with horizon intervals each hour, day and week. Results show an improvement in the estimated costs of running ambulance services from the dynamic model when compared with the static model.

Section 7.1 discusses the developments in the dynamic model that are extensions of the static model, including new assumptions; Section 7.2 presents a new MIP formulation, formulated with FJSS techniques; Section 7.3 discusses the solution approach for the MIP model, using one week of deterministic data as per the case study; Section 7.4 presents the results and sensitivity analysis; Section 7.5 explores a variation of the dynamic model, with an objective of minimising response times with a fixed shift schedule; and Section 7.6 discusses implications and further work.

7.1 EXTENSIONS TO THE PREVIOUS MODEL

The dynamic model introduces new parameters, variables and constraints to extend the static model. Some of these are to allow the model to be solved as a rolling horizon. A rolling horizon approach is suitable for the dynamic model to allow a large problem to be broken up into a series of smaller problems, each of which is easier and faster to solve. Each horizon covers a time interval which is a

subset of the total time period covered in the model and introduces new incidents arising during that time period. Additionally, selected incidents from the previous horizon are rolled over so that each horizon after the first is informed by any incidents which are still in progress from the previous horizon.

Relocations and reassignments are allowed in the model. Reassignments, where an incident or relocation was previously allocated an ambulance (or hospital allocation for incidents only) and is now assigned to a different one, can occur each time information is updated. For this model, information only updates at each new horizon with the introduction of new incidents. Relocations can occur at any time when an ambulance is available. The purpose of these jobs is to move an ambulance from one location to any of the ambulance stations in the model in order to be prepared for future incidents. Deterministic data informs relocations in the model, although coverage is used to inform relocations in a real time model presented in the next chapter.

A key component of the dynamic model is the inclusion of jobs returning ambulances to their home stations at the end of each shift. This allows overtime to be optimised in the objective function, as the clear time of the last job on each shift is the clear time of an appropriate return-to-station job, determined through precedence constraints on disjunctive variables involving the return-to-station jobs.

Additional shift scheduling rules are introduced into the model to take a further step towards realistic shift scheduling for ambulance crews within a single, integrated model. The minimum time off between shifts and preference for forward rostering present in the static model is continued in the dynamic model. The dynamic model also contains a limit on the number of consecutive night shifts, a limit on the maximum number of shifts per week, and a requirement for a rostered days off (RDO) period, comprising two full days in a row every week.

7.1.1 Assumptions

Assumptions in the dynamic model are similar to the static model but some have been relaxed. Assumptions which vary to those in the previous model are listed here and explained below:

- Pre-emption is permitted, but only during certain operations

- All business rules for crew scheduling must be obeyed
- Ambulances may be assigned to wait at any ambulance station

Incidents are assumed to all require processing of operations in the same order. These are: preparation required prior to an ambulance beginning travel; travelling to the scene of an incident; dealing with the incident at the scene; travelling to a secondary location (that is, a hospital) if necessary; ramping and admission time spent at a hospital; and any cleaning necessary after an incident has been completed on an ambulance. While ambulances are travelling or unassigned to an incident, they may be redirected to other locations. Empty ambulances may be reassigned to a higher priority incident than their current assignment, and ambulances transferring patients to a hospital may be redirected to another hospital if expected processing times change while en route. Pre-emption of an incident is only allowed during the first operation. Each job is restricted to only one ambulance at a time. Incidents with multiple responding ambulance units may be desirable in reality but are treated by this model as separate incidents with appropriate due dates and on-scene processing requirements. As with the static model, the term ambulance is used to refer to any ambulance vehicle with an appropriate ambulance crew. Ambulance IDs refer to the ambulance crew and not the vehicle.

This model is tested with deterministic data. Redirection to a different hospital is not expected to occur because processing times will not change. Reassignment of ambulances to different incidents may occur as new incidents are introduced in each new window in the rolling horizon. Available ambulances may also be redirected to travel to a new ambulance station if such a move would improve readiness for subsequent incidents or overtime at the end of a shift. Figure 7-1 shows an example of these processes from the point of view of an ambulance, showing location and activity.

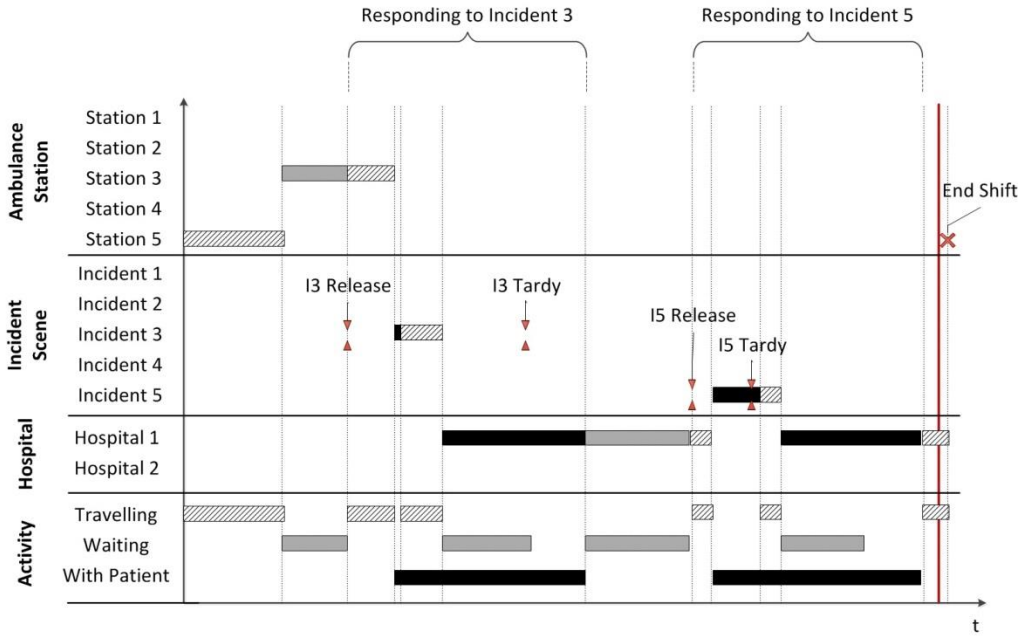


Figure 7-1 Example schedule of processes for two incidents on the same ambulance

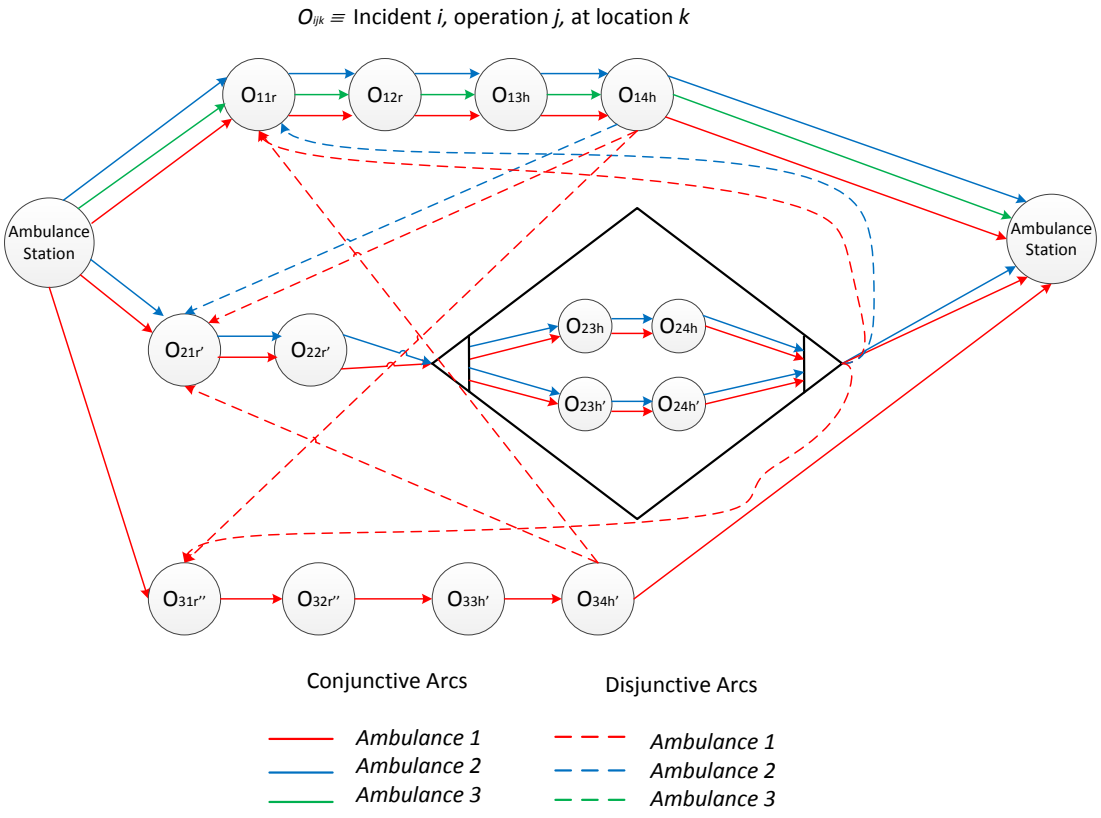


Figure 7-2 Example disjunctive graph for the dynamic model with three incidents, three ambulances and two hospitals

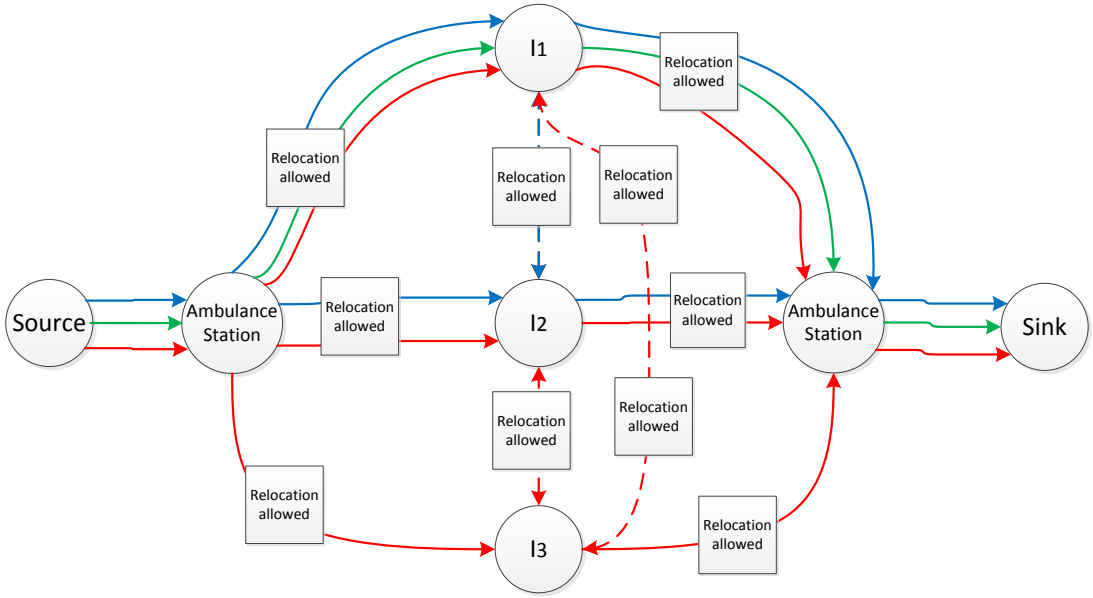


Figure 7-3 A sample presentation of sites at which relocations are available in the disjunctive graph for the dynamic model

7.1.2 Disjunctive Graph Representation

Disjunctive graph models are used in job shop problems to create an order of operations that are processed on the same machine. A disjunctive graph is a directed graph representing the possible schedules for all operations on all machines (Błażewicz et al., 2000; Brucker & Knust, 2006; Pinedo, 2012). Each node on the graph represents an operation on a machine (including dummy operations *0* and *n+1* to represent a source and a sink). Nodes representing operations with fixed precedence constraints are connected by solid conjunctive arcs. The set of disjunctive constraints, connecting all operations that are processed on the same machine, are represented by pairs of dashed lines. Each arc is weighted by the processing time of the node at the beginning of the arc. Błażewicz et al. (2000) present a representation of the disjunctive graph that aims to increase efficiency by specifying the relationship between two tasks on the graph as unknown, successor or predecessor. Solving the disjunctive graph involves selecting one disjunctive arc from each pair of disjunctive constraints.

Disjunctive constraints are used both to prevent overlapping processing times and to prevent overlapping locations. This approach ensures that an ambulance cannot be in two places as once and must have sufficient travel time for a change in

location to occur. Location disjunctions are used for jobs introduced to relocate ambulances during a shift and to return ambulances to the correct station at the end of a shift.

A novel precedence constraint on the disjunctive variables is also introduced. The new constraint forces the last incident on each shift, whenever an ambulance is assigned to that shift, to be a return-to-station job appropriate for that ambulance. This allows the time at which an ambulance arrives at its home station at the end of the shift to be easily extracted and so it becomes easy to calculate overtime within the model. Figure 7-2 is an example disjunctive graph for three incidents where Incident 1 can use any of three ambulances, Incident 2 can use only two of the three ambulances and Incident 3 is limited to one specific ambulance. Additionally, Incident 2 has a choice of two hospitals while Incidents 1 and 3 are restricted to one. Conjunctive arcs are represented by solid lines and disjunctive arcs, between incidents, by dashed lines. Each ambulance is represented by a different colour. Where a decision arc exists such that only one of several arcs will be traversed, this is represented in the diagram though a split in the paths which converge again at the completion of the relevant operations.

Figure 7-3 collapses the operations within each incident (now represented by I_1, I_2 and I_3) and indicates the points at which relocations are possible. Relocations may occur at any point before and after an incident but will always be directed toward an ambulance station. A feasible schedule, including relocations and all operations for three incidents, is presented in Figure 7-4. Relocations are indicated as (R_s^r) where s indicates the ambulance station that is destination of the relocation and r is a unique identifier for relocations. For a feasible schedule, constraints on ambulances require that the first location attended by an ambulance after the source must be the same as the final location attended by the ambulance prior to the sink.

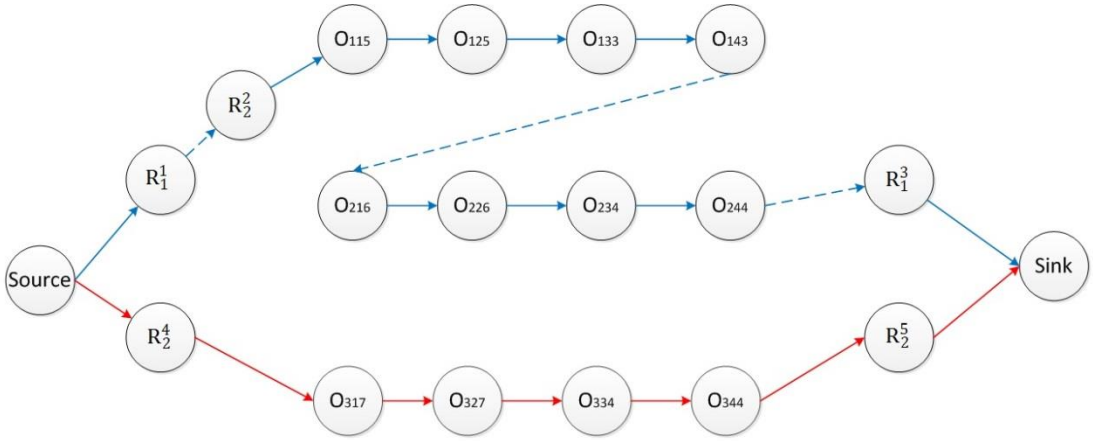


Figure 7-4 Example feasible schedule for the dynamic model

7.2 FORMULATION

The dynamic model presented in this chapter is an extension of the FFSS model developed in the previous chapter. The static dispatching condition has been improved to allow dynamic dispatching and relocation of ambulances. Shift scheduling rules are integrated within a single model so that, when the model is solved on weekly scales, conditions on the number of shifts per week, consecutive night shifts and time off requirements are considered. The model seeks to optimise this shift schedule while meeting demand through dynamic scheduling. It is different to previous dynamic models in the literature which consider the shifts based on demand profiles and then seek to optimise relocations during each shift in a second model. Return-to-station jobs at the end of each shift are introduced to allow overtime to be considered. These are managed by establishing new parameters, variables and constraints. An overview of the model is shown in Figure 7-5.

7.2.1 Parameters

Time step parameters are required to define each horizon for the rolling horizon approach.

t	Time at which rolling horizon begins
t_{step}	Time interval between each horizon
$Tmax$	Number of time steps

Figure 7-5 Outline of the dynamic model with input, output and solution approach

Static parameters exist to define the hospitals, ambulances and ambulance stations present. Sets of parameters are shown in bold. Shift scheduling rules in the dynamic model also require sets to be defined for overnight shifts and shifts each week.

Wmax	Total number of weeks covered
W	Set of all weeks {1..Wmax}

Fmax	Total number of shifts
$\boldsymbol{F}$	Set of all shifts {1..Fmax}
$\boldsymbol{F_w}$	Set of all shifts beginning in week *w*
$\boldsymbol{G}$	Set of all night shifts
B_f	Beginning time of shift *f*
E_f	Ending time of shift *f*
Hmax	Total number of hospitals
$\boldsymbol{H}$	Set of all hospitals {1..Hmax}
Smax	Total number of ambulance stations
$\boldsymbol{S}$	Set of all ambulance stations {1..Smax}
Kmax	Total number of different ambulance vehicle types
$\boldsymbol{K}$	Set of different ambulance vehicle types {1..Vmax}
Amax	Total number of ambulances in the system
$\boldsymbol{A}$	Set of all ambulances {1..Amax} in the system
$\boldsymbol{\Lambda_k}$	Set of all ambulances of vehicle type *k*
Pmax	Total number of incident priority types (i.e. triage categories)
$\boldsymbol{P}$	Set of all priority types {1..Pmax}
M	Large value for logical constraints: $M = 2 \times E_{Fmax}$
ω_k	Weights applied to each ambulance type *k* to represent cost in the objective function
σ_k	Weights applied to each ambulance type *k* to represent cost of overtime in the objective function

Incidents, potential relocation and return-to-station jobs must each be defined as separate sub-sets from the set of all jobs present in the model at each time step. At the end of each horizon, relocation and return-to-station jobs that have begun are treated as incidents, with appropriate parameters, in the next horizon.

Jmax(t)	Total number of jobs at time *t*
$\boldsymbol{J(t)}$	Set of all jobs {1..Jmax} at time *t*
Imax(t)	Total number of incidents at time *t*
$\boldsymbol{I(t)}$	Set of all incidents (1..Imax} at time *t*
$\boldsymbol{J^S(t)}$	Set of all potential jobs returning ambulances to home stations at time *t*
$\boldsymbol{J^R(t)}$	Set of all potential relocation jobs at time *t*

Location parameters for ambulance stations and hospitals are static, however, locations for incidents depend on which incidents are present in each horizon.

Nmax(t)	Total number of location nodes at time *t*
N(t)	Set of all location nodes at time *t*
$\boldsymbol{N_H}$	Set of all location nodes covering hospitals
$\boldsymbol{N_S}$	Set of all location nodes covering stations
$\boldsymbol{N_I(t)}$	Set of all location nodes covering incident locations at time *t*
$\mu_{l_1 l_2}(t)$	Expected travel time from location l_1 to l_2 at time *t*

Job parameters vary for each horizon as the incidents vary. The different processes for each incident are now also separated into a larger number of operations for the dynamic model than were present in the static model. This is because some of these processes may now be interrupted for reassignments. For relocation and return-to-station jobs, many of these values will be zero. Ambulances that are suitable to respond to jobs are now no longer just about appropriate vehicle types, as only the correct return-to-station job can be used to return ambulance *a* to an ambulance station.

$R_i(t)$	Release time of incident *i* at time *t*
$L_i(t)$	Location of incident *i* at time *t*
$T_i(t)$	Tardy response time for incident *i* at time *t*
$U_i(t)$	Upper bound on arrival time for incident *i* at time *t*
$P_{ip}(t)$	$= \begin{cases} 1, \text{ if incident } i \text{ has priority type p at time } t, \\ 0, \text{ otherwise} \end{cases}$
$Q_p(t)$	Maximum number of incidents of priority type *p* that can be tardy at time *t*
$\pi_{ih}(t)$	Hospital requirements for incident *i* at time *t*
$\alpha_{ia}(t)$	Expected time for ambulance *a* to prepare for a response to incident *i* at time *t*
$\gamma_{ia}(t)$	Expected time for ambulance *a* to handle incident *i* on site at time *t*
$\zeta_{ih}(t)$	Expected time that incident *i* will spend ramping at hospital *h* at time *t*
$\eta_{ih}(t)$	Expected time for incident *i* to be passed onto/admitted into hospital *h* at time *t*

$\lambda_{ia}(t)$ Expected time for cleaning ambulance a after responding to incident i at time t

$\xi_{ja}(t)$ $= \begin{cases} 1, & \text{if ambulance } a \text{ is suitable to respond to job } j \text{ at time } t, \\ 0, & \text{otherwise} \end{cases}$

7.2.2 Variables

7.2.2.1 Decision Variables

The three binary decision variables introduced in the static model are present in the dynamic model, but have been adapted to enable them to take on different values at different time steps.

$$x_{iaf}(t) = \begin{cases} 1, & \text{if incident } i \text{ is assigned to ambulance } a \text{ during shift } f \text{ at time } t, \\ 0, & \text{otherwise} \end{cases}$$

$$y_{ih}(t) = \begin{cases} 1, & \text{if incident } i \text{ is assigned to hospital } h \text{ time } t, \\ 0, & \text{otherwise} \end{cases}$$

$$v_{asf}(t) = \begin{cases} 1, & \text{if ambulance } a \text{ to work shift } f, \text{with home station } s, \text{at time } t, \\ 0, & \text{otherwise} \end{cases}$$

7.2.2.2 Dependent Variables

Dependent variables for dispatch, arrival and clear times are adapted from the static model and made dynamic to allow new values for each horizon. Additional dynamic variables for the time at which ambulances depart for or arrive at hospitals have now been included in case reassignment to a different hospital is a preferable solution once new information becomes available. These variables may take on continuous values.

$d_{ia}(t)$ Expected dispatch time for job i on ambulance a during time step t

$r_{ia}(t)$ Expected arrival time for job i on ambulance a during time step t

$c_{ia}(t)$ Expected clear time for job i on ambulance a during time step t

$e_{ih}(t)$ Expected time for job i to start travel toward hospital h

$g_{ih}(t)$ Expected time for job i to arrive at hospital h

The overtime variable is also extended to be dynamic. As before, it may take on continuous values.

$o_{af}(t)$ Overtime accrued by ambulance a during shift f at time step t

The tardy variable remains and become dynamic.

$$q_i(t) = \begin{cases} 1, & \text{if incident } i \text{ is tardy,} \\ 0, & \text{otherwise} \end{cases}$$

The disjunctive variable from the static model becomes dynamic and is joined by a new disjunctive variable for immediate predecessors. Both are necessary as the immediate predecessor informs location disjunctions while the original disjunctive variable contains information about the ambulance and shift which two incidents share.

$$z_{ijaf}(t) = \begin{cases} 1, & \text{if incident } i \text{ precedes incident } j \text{ on ambulance } a \text{ during shift } f, \\ 0, & \text{otherwise} \end{cases}$$

$$Z_{ij}(t) = \begin{cases} 1, & \text{if incident } i \text{ immediately precedes incident } j \\ & \quad \text{on the same ambulance and shift,} \\ 0, & \text{otherwise} \end{cases}$$

New dependent variables related to locations are introduced, one depending on the location at which an ambulance was at prior to responding to an incident, and the second for the location at which an ambulance became clear after handling an incident.

$$\delta_{in}(t) = \begin{cases} 1, & \text{if incident } i \text{ received dispatch from node } n, \\ 0, & \text{otherwise} \end{cases}$$

$$\theta_{in}(t) = \begin{cases} 1, & \text{if incident } i \text{ clears node } n, \\ 0, & \text{otherwise} \end{cases}$$

New shift scheduling rules added in the dynamic model require one new variable to be established to identify the first shift of the RDO for an ambulance crew.

$$\psi_{af}(t) = \begin{cases} 1, & \text{if shift } f \text{ is the first shift of RDO period for ambulance } a, \\ 0, & \text{otherwise} \end{cases}$$

Finally, new dependent variables are introduced to indicate incident status for the rolling horizon. These determine which constraints will apply to these incidents and ambulances responding to these incidents.

$I^1(t)$ Set of all incidents known but where response is not yet on the way

$I^2(t)$ Set of all incidents with a response travelling toward the site

$I^3(t)$ Set of all incidents with a response currently active on site

$I^4(t)$ Set of all incidents with a response complete and ambulance en route to a hospital

$I^5(t)$ Set of all incidents currently at hospital

$I^6(t)$ Set of all incidents fully cleared.

7.2.3 Objective

The objective, similar to the model presented in the previous chapter, is to minimise the expected costs of running an ambulance service. This is done through minimising the number of shifts where ambulances are working, minimising overtime and selecting, where possible, the cheapest ambulances to schedule. The binary variable representing the shift onto which each ambulance is scheduled, and the continuous overtime variable for each ambulance and shift, form part of the weighted sum, shown below, to be minimised.

Minimise

$$\sum_{k \in \boldsymbol{K}} \left(\omega_k \sum_{a \in \boldsymbol{A_k}} \sum_{s \in \boldsymbol{S}} \sum_{f \in \boldsymbol{F}} v_{a_k s f}(t) + \sigma_k \sum_{a \in \boldsymbol{A_k}} \sum_{f \in \boldsymbol{F}} o_{af}(t) \right)$$

7.2.4 Constraints

Precedence Constraints

Precedence constraints in the dynamic model include the precedence constraints from the static model, with the variables now as dynamic variables, and extend these constraints to be relevant for relocation and return-to-station jobs. Precedence constraints on the decision variables now have to consider the time at which a decision is made so that any new decisions cannot take effect any earlier than the time *t* that the current horizon of the rolling horizon began. Decisions made in a previous horizon must be allowed to continue with the same time as they had previously. The full set of precedence constraints is described below.

Constraint (7.1): Ambulances cannot be dispatched to incidents prior to the release time of each incident:

$$d_{ia}(t) + M \left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \right) \geq R_i \qquad \forall i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.1}$$

Constraint (7.2): Jobs, inclusive of incidents, relocations and return-to-station jobs, cannot be dispatched during a shift until that shift has commenced:

$$d_{ia}(t) + M \left(1 - x_{iaf}(t) \right) \geq B_f \qquad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.2}$$

Continuity between horizons must also be respected with constraints on dispatch times.

Constraint (7.3): Where dispatch occurred before time *t*, and the response continues on the same ambulance after time *t*, the dispatch time for the horizon beginning at time *t* must remain the same as the dispatch time from the previous horizon:

$$d_{ia}(t) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t)\right) \geq d_{ia}\left(t - T_{step}\right) - M * I_i^1(t) \qquad \begin{matrix} \forall\, i \in \boldsymbol{J}, \\ a \in \boldsymbol{A} \end{matrix} \qquad (7.3)$$

Constraint (7.4): If a reassignment decision is made during a horizon beginning at time *t*, the dispatch time on the new ambulance must be greater than or equal to *t*. Reassignment may only occur for incidents that have not yet reached their intended destination:

$$\begin{aligned} d_{ia}(t) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t)\right) & \qquad \begin{matrix} \forall\, i \in \boldsymbol{I}, \\ a \in \boldsymbol{A}, \\ a' \in \boldsymbol{A} \,/\, \{a\} \end{matrix} \\ \geq t\left(\sum_{f \in \boldsymbol{F}} x_{ia'f}(t - T_{step})\right) - M\left(1 - I_i^2(t)\right) & \end{aligned} \qquad (7.4)$$

Constraint (7.5): Any job where dispatch has not occurred prior to the beginning of a horizon has a dispatch time greater than or equal to *t* on any ambulance to which it is assigned in this horizon:

$$d_{ia}(t) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t)\right) \geq t - M\left(\sum_{f \in \boldsymbol{F}} x_{iaf}\left(t - T_{step}\right)\right) \qquad \begin{matrix} \forall\, i \in \boldsymbol{J}, \\ a \in \boldsymbol{A} \end{matrix} \qquad (7.5)$$

Constraint (7.6): The expected arrival time of each incident (arrival of assigned ambulance at the scene of an incident) must be greater than or equal to the time of dispatch plus the time for preparation and travel:

$$\begin{aligned} r_{ia}(t) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t)\right) & \qquad \begin{matrix} \forall i \in \boldsymbol{J}, \\ a \in \boldsymbol{A} \end{matrix} \\ \geq d_{ia}(t) + \alpha_{ia}(t) + \sum_{n \in \boldsymbol{N}} \delta_{in}(t)\mu_{nl_i}(t) & \end{aligned} \qquad (7.6)$$

Constraint (7.7): Transfer of an incident to a hospital cannot begin prior to that incident being ready for transfer, i.e. prior to the conclusion of treatment at the scene which began upon arrival:

$$e_{ih}(t) + M\big(1 - y_{ih}(t)\big) \geq r_{ia}(t) + \gamma_{ia}(t) \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \quad \forall i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.7}$$

Constraint (7.8): Expected time of arrival at a hospital for each incident must be greater than or equal to the time that transfer to the hospital commenced plus travel time to the hospital from the scene:

$$g_{ih}(t) + M\big(1 - y_{ih}(t)\big) \geq e_{ih}(t) + \mu_{l_i l_h} \quad \forall i \in \boldsymbol{I}, h \in \boldsymbol{H} \tag{7.8}$$

Continuity constraints, similar to those for dispatch times, apply for the time at which incidents requiring transportation to a hospital begin transfer. Reassignment of hospitals is possible up until the incident arrives at the hospital.

Constraint (7.9): If transport to a hospital began prior to time *t*, and no reassignment has occurred, then time for the start of transportation to hospital for the horizon beginning at time *t* must remain the same as from the previous horizon:

$$e_{ih}(t) + M\big(1 - y_{ih}(t)\big) \geq e_{ih}\big(t - T_{step}\big) - M\left(1 - \sum_{m=4}^{6} I_i^m(t)\right) \quad \begin{matrix} \forall\, i \in \boldsymbol{I}, \\ h \in \boldsymbol{H} \end{matrix} \tag{7.9}$$

Constraint (7.10): If a decision is made during the horizon beginning at time *t* to reassign an incident to a new hospital then the time at which the incident begins travel to the new hospital cannot be earlier than time *t*:

$$\begin{aligned} e_{ih}(t) + M\big(1 - y_{ih}(t)\big) & \qquad \forall\, i \in \boldsymbol{I}, h \in \boldsymbol{H}, \\ \geq t\left(y_{ih'}(t - T_{step})\right) - M\left(1 - I_i^4(t)\right) & \qquad h' \in \boldsymbol{H} \,/\, \{h\} \end{aligned} \tag{7.10}$$

Clear time for incidents is expressed through two linear constraints that constrain clear time when hospital transfer is or is not required. The first of these constraints is relevant for relocation and return-to-station jobs as well.

Constraints (7.11): Expected clear time for each job must be greater than or equal to the time that the response arrived, plus the appropriate amount of time spent on scene and cleaning the ambulance afterward:

$$c_{ia}(t) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t)\right) \geq r_{ia}(t) + \gamma_{ia}(t) + \lambda_{ia}(t) \quad \begin{matrix} \forall\, i \in \boldsymbol{J}, \\ a \in \boldsymbol{A} \end{matrix} \tag{7.11}$$

Constraint (7.12): Expected clear time for any incident transferred to a hospital must be greater than or equal to the time arrived at the hospital plus time spent at hospital and any cleaning time:

$$c_{ia}(t) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t)\right) \geq \sum_{h \in \boldsymbol{H}} g_{ih}(t) + y_{ih}(t)\big(\zeta_{ih}(t) + \eta_{ih}(t)\big) + \lambda_{ia}(t) \qquad \forall\, i \in \boldsymbol{I},\ a \in \boldsymbol{A} \tag{7.12}$$

Constraints (7.13) through to (7.17): The following constraints are required to set dependent variables d_{ia}, r_{ia}, c_{ia}, e_{ih} and g_{ih} equal to zero if there is no assignment:

$$d_{ia}(t) \leq M \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \qquad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{7.13}$$

$$r_{ia}(t) \leq M \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \qquad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{7.14}$$

$$c_{ia}(t) \leq M \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \qquad \forall\, i \in \boldsymbol{J}\ a \in \boldsymbol{A} \tag{7.15}$$

$$e_{ih}(t) \leq M y_{ih}(t) \qquad \forall\, i \in \boldsymbol{J}, h \in \boldsymbol{H} \tag{7.16}$$

$$g_{ih}(t) \leq M y_{ih}(t) \qquad \forall\, i \in \boldsymbol{J}, h \in \boldsymbol{H} \tag{7.17}$$

Disjunctive Constraints

These constraints are required in the scheduling model to ensure that there is no overlap between any jobs, inclusive of incidents, relocations and return-to-station jobs.

Constraints (7.18) and (7.19): A pair of disjunctive sequencing logical constraints ensures that there is no overlap between jobs assigned to the same ambulance, that is, the clear time of the earlier job must be earlier than or at the same time as the dispatch time of the later job:

$$d_{ia}(t) - c_{ja}(t) + 2M(1 - x_{iaf}(t)) + 2M(1 - x_{jaf}(t)) \geq M\big(z_{jiaf}(t) - 1\big) \qquad \forall\, i \in \boldsymbol{J},\ j \in \boldsymbol{J}/\{i\},\ a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.18}$$

$$d_{ja}(t) - c_{ia}(t) + 2M(1 - x_{iaf}(t)) + 2M(1 - x_{jaf}(t)) \geq -M z_{jiaf}(t) \qquad \forall\, i \in \boldsymbol{J},\ j \in \boldsymbol{J}/\{i\}, \tag{7.19}$$

$$a \in \boldsymbol{A}, f \in \boldsymbol{F}$$

Constraints (7.20) and (7.21): The disjunctive variables for jobs *i* and *j* on ambulance *a* and shift *f* should equal zero if either job is not assigned to the ambulance or shift in question:

$$z_{ijaf}(t) \leq x_{iaf}(t) \qquad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J}\,/\{i\}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.20}$$

$$z_{ijaf}(t) \leq x_{jaf}(t) \qquad \forall\, i \in \boldsymbol{J}\,, j \in \boldsymbol{J}\,/\{i\}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.21}$$

Constraint (7.22): A disjunctive variable will also be equal to zero along the diagonal as it is illogical for a job to precede itself:

$$z_{iiaf}(t) = 0 \qquad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.22}$$

Constraint (7.23): A disjunctive precedence relation is applied to all return-to-station jobs. This constraint ensures that the last job for an ambulance on every assigned shift will return the ambulance to an ambulance station:

$$z_{ijaf}(t) \geq x_{jaf}(t) - M\big(1 - x_{iaf}(t)\big) \qquad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J^S} \setminus \{i\}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.23}$$

The following constraints relate the dependent immediate predecessor disjunctive variable to the original disjunctive decision variable.

Constraint (7.24): The immediate predecessor variable must be zero where there are no predecessors:

$$Z_{ij}(t) \leq \sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} z_{ijaf}(t) \qquad \forall\, i \in \boldsymbol{J}\,, j \in \boldsymbol{J} \setminus \{i\} \tag{7.24}$$

Constraint (7.25): If there is at least one predecessor then there must be an immediate predecessor:

$$J_{max} \sum_{j \in \boldsymbol{J}} Z_{ij}(t) \geq \sum_{j \in \boldsymbol{J}} \sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} z_{ijaf}(t) \qquad \forall\, i \in \boldsymbol{J} \tag{7.25}$$

Constraints (7.26) and (7.27): There must be at most one immediate predecessor and one immediate antecedent for each job:

$$\sum_{j \in \boldsymbol{J}} Z_{ij}(t) \leq 1 \qquad \forall\, i \in \boldsymbol{J} \tag{7.26}$$

$$\sum_{j \in \boldsymbol{J}} Z_{ji}(t) \leq 1 \qquad \forall\, i \in \boldsymbol{J} \tag{7.27}$$

Resource Constraints

The following constraints limit the ambulances that are allowed to be assigned to jobs. Each job is restricted to a single ambulance each horizon. By restricting each

job to a single ambulance each horizon, each job is also restricted to a single dispatch time. Incidents must be assigned an ambulance each horizon; potential relocation and return-to-station jobs should only be assigned where necessary.

Constraint (7.28): Each incident must be assigned exactly one ambulance, during exactly one shift:

$$\sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} x_{iaf}(t) = 1 \quad \forall\, i \in \boldsymbol{I} \tag{7.28}$$

Constraint (7.29): Potential relocation and return-to-station jobs can have, at most, one ambulance and shift:

$$\sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \leq 1 \quad \forall\, i \in \{\boldsymbol{J^R}, \boldsymbol{J^S}\} \tag{7.29}$$

Constraint (7.30): All incidents must be handled by appropriate ambulances. Jobs introduced to return ambulance crews to their home station are made to return the correct ambulance by controlling the input parameter for appropriate ambulances and applying the following constraint:

$$\sum_{f \in \boldsymbol{F}} x_{iaf}(t) \leq \xi_{ia}(t) \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{7.30}$$

Constraint (7.31): Reassignment to a difference ambulance is forbidden once the ambulance has arrived at the scene of an incident. This constraint forces the incident to continue with the same ambulance:

$$x_{iaf}(t) \geq x_{iaf}\left(t - T_{step}\right) - M\left(1 - \sum_{m=3}^{6} I_i^m(t)\right) \quad \forall\, i \in \boldsymbol{I}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.31}$$

Constraint (7.32): Continuity for ambulance crews requires that ambulance crews assigned to shifts in previous horizons must still be considered to be assigned to those shifts in later horizons. New shifts may be added dynamically but cannot be discarded after they have been utilised:

$$v_{asf}(t) \geq v_{asf}\left(t - T_{step}\right) \quad \forall\, a \in \boldsymbol{A}, s \in \boldsymbol{S}, f \in \boldsymbol{F} \tag{7.32}$$

During each horizon, each incident may only be assigned to a single hospital, appropriate to the requirements set in the input parameters. Relocation and return-to-station jobs are never sent to hospitals.

Constraint (7.33): Each incident can be assigned to at most one hospital at a time:

$$\sum_{h \in \boldsymbol{H}} y_{ih}(t) \leq 1 \quad \forall\, i \in \boldsymbol{I} \tag{7.33}$$

Constraint (7.34): Each incident must be assigned a hospital where transport to a hospital is required:

$$M \sum_{h \in \boldsymbol{H}} y_{ih}(t) \geq \sum_{h \in \boldsymbol{H}} \pi_{ih}(t) \quad \forall\, i \in \boldsymbol{I} \tag{7.34}$$

Constraint (7.35): Each incident can only be sent to appropriate hospitals. Relocation and return-to-station jobs have no appropriate hospitals to which they can be sent and should never be assigned to a hospital:

$$y_{ih}(t) \leq \pi_{ih}(t) \quad \forall\, i \in \boldsymbol{J}, h \in \boldsymbol{H} \tag{7.35}$$

Constraint (7.36): Once an incident has arrived at a hospital, the assignments for each subsequent horizon must continue the assignment from the previous horizon:

$$y_{ih}(t) \geq\ y_{ih}\left(t - T_{step}\right) -\ M\left(1 - \sum_{m=5}^{6} I_i^m(t)\right) \quad \forall\, i \in \boldsymbol{I}, h \in \boldsymbol{H} \tag{7.36}$$

Tardy constraints

Tardy constraints apply to incidents. Every incident has two due dates (T_i and U_i) constraining arrival times (i.e. $r_{ia}(t)$). These ensure that ambulances arrive on the scene of the incident quickly enough to meet performance measure requirements for every horizon.

Constraint (7.37): All incidents must receive a response by the upper due date:

$$r_{ia}(t) \leq U_i \quad \forall\, i \in \boldsymbol{I} \tag{7.37}$$

Constraint (7.38): An incident is considered tardy if a response is received after the tardy due date:

$$Mq_i(t) \geq r_{ia}(t) - T_i(t) \quad \forall\, i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.38}$$

Constraint (7.39): The number of tardy incidents is limited for each priority type:

$$\sum_{i \in \boldsymbol{I}} q_i(t)\ P_{ip}(t) \leq Q_p(t) \quad \forall\, p \in \boldsymbol{P} \tag{7.39}$$

Overtime/return-to-station constraints

Ambulance crews must end each shift at the station where they began that shift. Jobs returning ambulances to home stations are introduced to achieve this and must be constrained appropriately. Disjunctive constraints ensure that return-to-station jobs occur last on every shift, therefore, it is possible to determine overtime from this subset of all jobs. Including overtime in the objective of a dynamic ambulance scheduling model, and the method of determining overtime, is one of the innovations of this formulation. Overtime accrues for each minute past the end time of an assigned shift that an ambulance crew is busy or in the process of returning to the appropriate ambulance station to complete a shift.

Constraint (7.40): Overtime accrued by each ambulance crew, on each shift, is greater than or equal to the clear time of the return-to-station job for that ambulance and shift minus the end time of the shift:

$$c_{ia}(t) - E_f - o_{af}(t) \leq M\big(1 - x_{iaf}(t)\big) \quad \forall\, i \in \boldsymbol{J^S}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.40}$$

Constraint (7.41): In order to prevent excessive amounts of overtime, only return-to-station jobs can commence after a shift has ended. While incidents and relocations can continue past the end of a shift, ambulances cannot be dispatched to new incidents or relocations after the end of their shift:

$$d_{ia}(t) - E_f \leq M\left(1 - x_{iaf}(t)\right) \quad \forall\, i \in \{\boldsymbol{I}, \boldsymbol{J^R}\}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.41}$$

Constraint (7.42): A job returning ambulance *a* to a station at the end of shift *f* must be utilised if there are any jobs at all using ambulance *a* during shift *f*:

$$J_{max} * \sum_{i \in \boldsymbol{J^S}} x_{iaf}(t) \geq \sum_{j \in \boldsymbol{J}} x_{jaf}(t) \quad \forall\, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.42}$$

Constraint (7.43): There can be, at most, one job returning ambulance *a* to a station at the end of shift *f*:

$$\sum_{i \in \boldsymbol{J^S}} x_{iaf}(t) \leq 1 \quad \forall\, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.43}$$

Constraint (7.44): A job returning an ambulance to a station at the end of a shift must clear at the correct location, i.e. the location of home station to which the ambulance was assigned. This will ensure the correct return-to-station job is selected out of the options available in the model:

$$\theta_{iN_S}(t) + M\left(1 - x_{iaf}(t)\right) \geq v_{asf}(t) \quad \forall i \in \boldsymbol{J^S}, a \in \boldsymbol{A}, f \in \boldsymbol{F}, s \in \boldsymbol{S}, \tag{7.44}$$

Shift Scheduling Constraints

A number of shift scheduling rules for ambulance crew have been integrated into the ambulance scheduling model as constraints. This concept was introduced in the static model where minimum time off between shifts was enforced and is now extended to include additional shift scheduling rules in the dynamic relocation model. The new rules impose: a maximum number of shifts per week; a maximum number of consecutive overnight shifts; and a weekly RDO period consisting of 48 hours, beginning and ending at midnight.

Constraint (7.45) An ambulance crew must be assigned onto shift *f*, at any station, if they are assigned to any incident on shift *f*:

$$\sum_{s \in \boldsymbol{S}} v_{asf}(t) \geq x_{iaf}(t) \quad \forall\, i \in \boldsymbol{I}, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.45}$$

Constraint (7.46): Ambulance crews are restricted to a single ambulance station across all shifts. The ambulance station to which an ambulance crew are assigned on their first shift is the station that they will begin and end every shift. It is possible to relax this constraint to allow ambulance crews to begin shifts at stations within a neighbourhood rather than only one station as an extension to this model:

$$v_{asf}(t) + v_{as'f'}(t) \leq 1 \quad \forall\, a \in \boldsymbol{A}, s \in \boldsymbol{S}, f \in \boldsymbol{F}, f' \in \boldsymbol{F} \tag{7.46}$$

Constraint (7.47): Ambulance crews must have at a minimum time off of at least two shifts in between scheduled shifts. As in the static model, this satisfies the requirement to have a minimum of 8 hours off between finishing a shift (or overtime) and beginning the next. It also enforces the practice of forward rostering where there has not been a full day of rest between shifts:

$$\sum_{s \in \boldsymbol{S}} v_{asf}(t) + \sum_{s \in \boldsymbol{S}} v_{as(f+1)}\,(t) + \sum_{s \in \boldsymbol{S}} v_{as(f+2)}\,(t) \leq 1 \quad \forall\, a \in \boldsymbol{A}, f \in \boldsymbol{F} \tag{7.47}$$

The following constraints apply to weekly shift schedules. These are new constraints for the integrated ambulance scheduling and shift scheduling model.

Constraint (7.48): Ambulance crews should be assigned to a maximum four shifts per week:

$$\sum_{f \in \mathbb{F}_{\boldsymbol{w}}} \sum_{s \in \boldsymbol{S}} v_{asf}(t) \leq 4 \qquad \forall\, a \in \boldsymbol{A}, w \in \boldsymbol{W} \tag{7.48}$$

Night shifts occur once per day, with two shifts in between. If shift *f* is a night shift then all shifts *(f+3n)* where *n* is an integer will also be night shifts. Consecutive night shifts can be limited using this inference, however, the form of a constraint on consecutive night shifts is dependent on the assumptions surrounding shift patterns. Extensions to this model with different shift patterns will need to modify constraint (7.49).

Constraint (7.49): Ambulance crews cannot work more than two night shifts in a row:

$$\sum_{s \in \boldsymbol{S}} v_{asf}(t) + \sum_{s \in \boldsymbol{S}} v_{as(f+3)}\,(t) + \sum_{s \in \boldsymbol{S}} v_{as(f+6)}\,(t) \leq 2 \qquad \forall\, a \in \boldsymbol{A}, f \in \boldsymbol{G} \tag{7.49}$$

Ambulance crews should have a RDO period of 2 entire days, from midnight to midnight, each week where no shifts are assigned. There are three shifts in each 24 hour period and the option of specifying 6 consecutive shifts to make up the requirement for 2 days off exists. However, this does not guarantee a full 48 hours between shifts and is insufficient to guarantee the time off period covers two days from midnight to midnight. Instead, three consecutive night shifts are used to cover a period from midnight through a full 48 hours wherein an ambulance cannot be assigned to new shifts.

Constraint (7.50): A RDO period off must commence at least once per week for each ambulance crew:

$$\sum_{f \in \mathbb{F}_{\boldsymbol{w}}} \psi_{af}(t) \geq 1 \qquad \forall\, a \in \boldsymbol{A}, w \in \boldsymbol{W} \tag{7.50}$$

Constraint (7.51): A period of two rostered days off commences at the first of three consecutive overnight shifts where an ambulance crew is not scheduled to any of those night shifts, nor any daytime shifts in between those shifts:

$$M\left(1 - \psi_{af}(t)\right) \geq \sum_{s \in \boldsymbol{S}} \sum_{f'=f}^{f+6} v_{asf'}\,(t) \qquad \forall\, a \in \boldsymbol{A}, f \in \boldsymbol{G} \tag{7.51}$$

Location Constraints

The dynamic model requires the location of ambulances to be present as dependent variables. This is due to the requirement to start and finish a shift at the same location, the capability of the dynamic model to relocate ambulances and travel times being dependent on location. Creative use of the disjunctive variables fixes an ambulance to be in only one location at a time. The following constraints apply to the location variables.

Constraint (7.52): Where a job (inclusive of all incident, relocation or return-to-station jobs) is preceded immediately by another job, the dispatch location of the new job is the same as the clear location of the previous one:

$$\delta_{in}(t) \geq \theta_{jn}(t) + M\big(Z_{ji}(t) - 1\big) \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\} \tag{7.52}$$

Constraint (7.53): The first job to which an ambulance is assigned on a shift will be dispatched from the location of the home ambulance station for that ambulance:

$$\delta_{iN_s}(t) + M \sum_{j \in \boldsymbol{J}} z_{jiaf}(t) \geq v_{asf}(t) - M\big(1 - x_{iaf}(t)\big) \quad \begin{matrix} \forall\, i \in \boldsymbol{J}, \\ a \in \boldsymbol{A}, f \in \boldsymbol{F} \end{matrix} \tag{7.53}$$

Constraint (7.54): Jobs which are utilised must have exactly one dispatch location. This applies to all incidents and any relocation or return-to-station jobs that are selected:

$$\sum_{n \in \boldsymbol{N}} \delta_{in}(t) = \sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \quad \forall\, i \in \boldsymbol{J} \tag{7.54}$$

Constraint (7.55): Similarly, jobs which are utilised must have exactly one clear location. This applies to all incidents and any relocation or return-to-station jobs that are selected:

$$\sum_{n \in \boldsymbol{N}} \theta_{in}(t) = \sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \quad \forall\, i \in \boldsymbol{J} \tag{7.55}$$

Constraint (7.56): For any incident where there is transport to a hospital, the clear location of the incident will be the location of the hospital:

$$\theta_{iN_h}(t) \geq y_{ih}(t) - M\left(1 - x_{iaf}(t)\right) \quad \forall\, i \in \boldsymbol{I}, a \in \boldsymbol{A}, f \in \boldsymbol{F}, h \in \boldsymbol{H} \tag{7.56}$$

Constraint (7.57): For any incident where there is no transport to a hospital, the clear location must be the location of the scene of the incident:

$$\theta_{iL_i}(t) \geq 1 - \sum_{h \in H} y_{ih}(t) \quad \forall\, i \in \boldsymbol{I} \tag{7.57}$$

Constraint (7.58): Relocation and return-to-station jobs have the assigned destination for that job (i.e. the station to which an ambulance will be directed defined in the parameters as L_i) as the clear location for each relocation or return-to-station job that is present in the solution:

$$\theta_{iL_i}(t) \geq \left(\sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \right) \quad \forall\, i \in \{\boldsymbol{J^R}, \boldsymbol{J^S}\} \tag{7.58}$$

Constraint (7.59): Relocation jobs from one location to the same location are to be prevented:

$$\theta_{in}(t) + \delta_{in}(t) \leq 1 \quad \forall\, i \in \boldsymbol{J^R}, n \in \boldsymbol{N} \tag{7.59}$$

Incident Set constraints

The following set of constraints ensures that the variables indicating incident status for each horizon are updated appropriately each time step.

Constraint (7.60): Where dispatch for incident *i* is yet to occur at the beginning of the horizon (i.e. time *t*), that status of the incident will be specified by $I_i^1(t) = 1$:

$$-M * I_i^1(t) \leq t - d_{ia}\left(t - T_{step}\right) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t - T_{step})\right) \quad \forall\, i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.60}$$

Constraints (7.61) and (7.62): For incidents where an ambulance has been dispatched prior to time *t* but a response is yet to be received on scene, the status of the incident is defined by $I_i^2(t) = 1$:

$$-M\left(\sum_{m=2}^{6} I_i^m(t)\right) \leq d_{ia}\left(t - T_{step}\right) - t + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t - T_{step})\right) \quad \forall\, i \in \boldsymbol{I},\ a \in \boldsymbol{A} \tag{7.61}$$

$$-M\left(\sum_{m=1}^{2} I_i^m(t)\right) \leq t - r_{ia}(t - T_{step}) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t - T_{step})\right) \quad \forall\, i \in \boldsymbol{I},\ a \in \boldsymbol{A} \tag{7.62}$$

Constraints (7.63) and (7.64): Where a response has been received for incident *i* and the ambulance is still at the scene at time *t*, the incident has status $I_i^3(t) = 1$:

$$-M\left(\sum_{m=3}^{6} I_i^m(t)\right) \leq r_{ia}(t - T_{step}) - t + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t - T_{step})\right) \quad \forall\, i \in \boldsymbol{I},\ a \in \boldsymbol{A} \tag{7.63}$$

$$-M\left(\sum_{m=1}^{3} I_i^m(t)\right) \leq t - e_{ih}(t - T_{step}) + M\left(1 - y_{ih}(t - T_{step})\right) \quad \forall\, i \in \boldsymbol{I},\ h \in \boldsymbol{H} \tag{7.64}$$

Constraints (7.65) and (7.66): Where transportation to a hospital has begun for incident *i* but the ambulance is yet to arrive at a hospital at time *t*, the incident has status $I_i^4(t) = 1$:

$$-M\left(\sum_{m=4}^{6} I_i^m(t)\right) \leq e_{ih}(t - T_{step}) + M\left(1 - y_{ih}(t - T_{step})\right) - t \quad \forall\, i \in \boldsymbol{I},\ h \in \boldsymbol{H} \tag{7.65}$$

$$-M\left(\sum_{m=1}^{4} I_i^m(t)\right) \leq t - g_{ih}(t - T_{step}) + M\left(1 - y_{ih}(t - T_{step})\right) \quad \forall\, i \in \boldsymbol{I},\ h \in \boldsymbol{H} \tag{7.66}$$

Constraints (7.67) and (7.68): After an incident *i* has arrived at a hospital but is not yet clear at time *t*, the incident status is $I_i^5(t) = 1$:

$$-M\left(\sum_{m=5}^{6} I_i^m(t)\right) \leq g_{ih}(t - T_{step}) + M\left(1 - y_{ih}(t - T_{step})\right) - t \qquad \forall\, i \in \boldsymbol{I},\ h \in H \tag{7.67}$$

$$-M\left(\sum_{m=1}^{5} I_i^m(t)\right) \leq t - c_{ia}(t - T_{step}) + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t - T_{step})\right) \qquad \forall\, i \in \boldsymbol{I},\ a \in \boldsymbol{A} \tag{7.68}$$

Constraint (7.69): Where an incident *i* has been cleared prior to time *t*, the incident has status $I_i^6(t) = 1$:

$$-M * I_i^6(t) \leq c_{ia}(t - T_{step}) - t + M\left(1 - \sum_{f \in \boldsymbol{F}} x_{iaf}(t - T_{step})\right) \qquad \forall\, i \in \boldsymbol{I},\ a \in \boldsymbol{A} \tag{7.69}$$

Constraint (7.70): Each incident can only have one status for each horizon:

$$\sum_{m=1}^{6} I_i^m(t) = 1 \qquad \forall i \in \boldsymbol{I} \tag{7.70}$$

Symmetry breaking constraints

Symmetry breaking constraints are added to the model to reduce the number of equivalent solutions.

Constraint (7.71): If duplicate ambulances exist in the pool of available ambulances, attempt to assign the one with the lower index first:

$$\sum_{s \in \boldsymbol{S}} \sum_{f \in \boldsymbol{F}} v_{asf}\,(t) \leq \sum_{s \in \boldsymbol{S}} \sum_{f \in \boldsymbol{F}} v_{a' sf} \qquad \forall k \in \boldsymbol{K}, a \in \Lambda_{\boldsymbol{k}}, a' \in \Lambda_{\boldsymbol{k}} : (a' = a + 1) \tag{7.71}$$

Constraint (7.72): Where multiple relocation jobs with the same destination are introduced, utilise the appropriate jobs with the lowest index first:

$$\sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} x_{iaf}(t) \leq \sum_{a \in \boldsymbol{A}} \sum_{f \in \boldsymbol{F}} x_{jaf}(t) \qquad \forall\, i \in \boldsymbol{J^R}, j \in \boldsymbol{J^R} : (j > i\ \&\ L_j = L_i) \tag{7.72}$$

Non-negativity and integer constraints

The following constraints are required in the model.

Constraint (7.73): All time stamps in the model should be greater than zero and less than the final time in the model:

$$0 \leq d_{ia}(t), r_{ia}(t), e_{ia}(t), g_{ia}(t),\ c_{ia(t)}, o_{af}(t) \leq E_{F_{\max}} \quad \forall\ i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.73}$$

Constraint (7.74): The following decision and dependent variables should be binary:

$$\begin{aligned} &x_{iaf}(t), y_{ih}(t), v_{asf}(t), z_{ijaf}(t), Z_{ij}(t), q_i(t), && \forall\ i \in \boldsymbol{J}, a \in \boldsymbol{A}, s \in \boldsymbol{S}, \\ &I_i^{\hat{i} \in \{1,2,3,4,5,6\}} \in \{0,1\} && h \in \boldsymbol{H}, f \in \boldsymbol{F} \end{aligned} \tag{7.74}$$

Constraint (7.75): Dispatch and clear locations can only take on the values of the nodes specified in the model:

$$0 \leq \delta_{in}(t), \theta_{in}(t) \leq N_{max} \qquad \forall\ i \in \boldsymbol{J}, n \in \boldsymbol{N} \tag{7.75}$$

7.3 SOLUTION APPROACH

The case study representing one week of incidents across the inner north region of the Brisbane metropolitan area is solved using a rolling horizon approach for the dynamic model. A rolling horizon approach is used to allow relocations and reassignments to occur and reduce the number of variables that need to be considered at once. By restricting the number of incidents being considered, the number of disjunctive variables is greatly reduced. This reduces the total size of the problems to be solved. However, shift scheduling rules consider ambulances across the entire week. Therefore, variables related to ambulances must contain all ambulances and not just the ambulances required in individual horizons. This limits ability to decrease the number of variables. Heuristics are introduced to find solutions in a reasonable amount of time. A Constructive Heuristic, a hybrid Tabu Search and Constructive Heuristic, an Ant Colony Optimisation heuristic and a hybrid ACO+CH heuristic are all considered.

7.3.1 Case Study

The case study described in Chapter 5, and used to solve the static model in Chapter 6, is used for solving the dynamic model. This presents a single, week long,

deterministic scenario over which the dynamic model is solved. Ambulance relocations in solutions of the dynamic model are based on actual incidents appearing in the case study. As a consequence, the ambulance schedule in solutions to the model is expected to vary if case studies change. However, as with the static model, the ambulance crew schedule is expected to be more robust. Each ambulance crew responds to multiple incidents per shift and is less affected by individual incidents.

A strategic ambulance crew schedule can be developed from the dynamic model using deterministic data from the case study. A tactical ambulance schedule, integrated with the ambulance crew schedule, can be explored through adding incidents into the model as information becomes available. Historical data is used to develop expected incidents for relocation decisions. This approach provides an indication of whether sufficient ambulance crews are currently scheduled to respond to demand or if additional ambulance crews should be added to the existing shift schedule.

Two additional case studies, generated by the same process as the first case study but not subject to the same analysis as in Chapter 5, are also tested. This allows for an exploration of the sensitivity of the model and solution algorithm to changes in demand. A final case study, exploring an increased demand (increasing the frequency of calls by 50%) is also tested. These additional case studies are only tested with the most promising solution algorithms.

In the remainder of this chapter, the case studies are used to solve the model across various horizons. These can vary in length to suit either a strategic or tactical approach.

7.3.2 Rolling Horizon

The model is initially solved as a strategic problem with a single horizon covering the entire week. Daily and hourly intervals between the beginnings of successive horizons are then explored. Results from different interval lengths are compared against the weekly horizon by examining the results of the final horizon in each solution.

Horizons commencing on the beginning and ending times of each shift are also considered. However, the overlap between shifts during the day would mean that the

interval between horizon starting times based on shift times would vary throughout each day. For simplicity, the model is initially tested with only horizon intervals of equal length throughout the week.

The overlap between horizons is allowed to vary in size. Each horizon, upon initialisation, engenders new incidents and may also call relevant data from one horizon onward into the next. Relevant data to be carried over and form the overlap between horizons includes: any incidents or other ambulance movements begun in one horizon but due to end in a later horizon, as well as any still waiting for an ambulance to be dispatched. Information on ambulances, such as home station, vehicle type, assigned shifts and overtime accrued, will always be carried through to the next horizon. The final solution to the model is informed from the ambulance data present after the final horizon is solved. Figure 7-6 shows the process diagram for the rolling horizon. A detailed algorithm for the rolling horizon process is presented in Figure 7-7.

The process begins by initialising any parameters which remain fixed for all horizons, such as ambulance station locations, and then solving for horizons one at a time. Incidents with a release date greater than the time t_h (time at beginning of horizon h) but less than or equal to time $t_h+\Delta t$ (time at end of horizon h) are added into the data when horizon h is initialised, and remain in the system until they are cleared and no longer necessary to inform the location of ambulances at the beginning of the next horizon. New incidents for horizon h then need to be added to the same list of incidents as any incidents or partially completed relocations from the prior horizon. This list of incidents, and associated parameters, is then passed onto a heuristic solver to search for solutions for horizon h. The parameters within the heuristics applied in each horizon are independent of those from prior horizons. That is, pheromone applied in horizon h does not influence decisions made in horizon $h+1$ and the tabu list is set to be the null set with each new horizon. Any of the heuristics described in the remainder of this section may be used with the rolling horizon approach. Ambulance crew schedules are built upon each horizon and, once an ambulance is scheduled, remain in the model for all horizons. If, after the solution for horizon h is saved, there are no more horizons to search, then the final crew schedule is the schedule at the end of horizon h. If there are more horizons to solve, then the jobs to be carried over to the next horizon must be extracted. Incomplete

incidents will always be carried through to the next horizon. Incomplete relocations and return-to-station jobs may be carried if they have already begun. Cleared jobs may also be selected to be carried to the next horizon if they are necessary to determine the location of an ambulance available at the beginning of the next horizon.

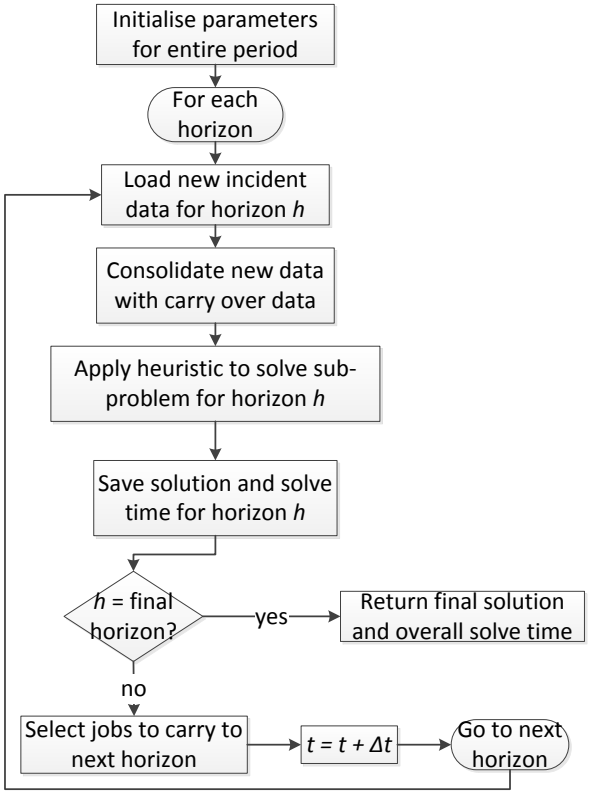


Figure 7-6 Rolling Horizon Process to solve the dynamic ambulance scheduling model

Rolling Horizon Algorithm

1: Initialise parameters for entire period
2: Initialise carry over data as empty sets and save as *Carry_Horizon*
3: Set t = t_0
3: **For** *h = 1 to num(Horizons)*
4: Extract and save new incident parameter data as *Horizon_Input*
5: **Consolidate horizon data**
6: Load *Carry_Horizon & Horizon_Input*
7: Convert relocation and return to station jobs from previous horizon into incidents with no due date
8: Combine new incident data with data from previous horizon
9: Update incident identities in disjunctive variable data
10: Save consolidated data as *Horizon Results*
11: **Run heuristic solver**
12: Update *Horizon Results*
13: Export objective function value, variables and solve time
14: **If** *h < num(Horizons)*
15: **Select Carry Horizon data**
16: Select data for incomplete incidents
17: Select data for complete immediate predecessors to incomplete incidents
18: Select data for incomplete relocation and return to station jobs *iff* the dispatch time is less than the beginning of the next horizon
19: Select complete return to station jobs *iff* the shift continues onto the next horizon
20: Update job reference identities in disjunctive variables
21: Select all ambulance data
22: **End If**
23: Update $t = t + \Delta t$
24: **End For**
25: Return solve time across all horizons

Figure 7-7 Algorithm to implement rolling horizons for the dynamic model

7.3.3 Constructive Heuristic

The CH for the dynamic model is strongly based on the CH for the static model. It has been adapted to allow for additional shift scheduling constraints, dynamic movements of ambulances and solving as a rolling horizon. As before, incidents are assigned to ambulances on a First Come First Served Basis (FCFS). Deterministic assignment places incidents onto existing ambulances and selects hospitals. Stochastic parameters are introduced to vary home ambulance stations, initial shift, and ambulance type whenever a new ambulance is required to be introduced into the system. New ambulances will only be introduced if there are no suitable ambulances available in the existing pool of ambulances. The CH also uses a cumulative distribution function to determine whether tardiness gets accepted. The process diagram and algorithm for this heuristic is shown in Figure 7-8.

This is an extension to the CH developed for the hybrid TS+CH in Section 6.2.3. Each incident is assigned to an ambulance in the order of arrival; however, incidents may have some variables fixed in the initialisation process if events have occurred prior to the time at which the horizon begins. The assignments required to be made for an incident in the CH depend on the state of the incident at time *t*, the beginning of the horizon. For incidents that have already arrived at hospital, no more decisions are required to be made and the variables are updated and saved for the current horizon. For incomplete incidents where an ambulance has arrived at the scene, but has not yet arrived at a hospital if hospital transportation is required, some variables are fixed but decisions on hospitals can be changed and updated. Feasible hospitals are identified and tested in order of shortest makespan. If disjunctive constraints indicate overlapping, the succeeding incident will have its variables cleared and be returned to the list of incidents for assignment.

Incidents where no response has yet arrived on scene follow a more complicated process. For each incident, parameters will determine if a tardy response can be accepted or not. The chance of later incidents accepting a tardy response shrinks each time a tardy response is accepted. All possible decision paths are then identified. This includes identifying all ambulances of a type suitable for the incident that are currently working on, or are allowed to be scheduled onto, shifts that cover the response time window for the incident. Incidents already assigned onto these ambulances and shifts are then identified and compared to the response time window for the current incident. Any other incident, already assigned onto a suitable ambulance for the current incident, may be a predecessor if their dispatch time is before the last time at which the current incident is allowed to receive a response. Incidents already assigned a suitable ambulance where dispatch is scheduled to occur after the release date of the current incident are possible antecedents. Some incidents may be identified as both a potential predecessor and antecedent if their dispatch time is within the time window between the release and upper due date for the current incident. Each pair of predecessors and antecedents defines a position in the ambulance schedule where the current incident may be placed. A position with no predecessor or no antecedent is also allowed. The combination of ambulance, shift, predecessor and antecedent is defined as a path for the current incident. These paths are explored, in order of earliest dispatch time until either a feasible assignment has

been found or no paths remain to explore. Relocations occurring between responses to incidents are explored and accepted only if they improve response times. In the event of no path returning a feasible solution, either due to tardiness or overlapping incidents, a new ambulance will be introduced into the system and assigned to the current incident.

Once all incidents have been assigned, return-to-station jobs are inserted into the schedule for all ambulances. For ambulances which were introduced into the system for the first time, a search is also performed to investigate whether overall overtime can be reduced without disrupting established arrival times, by modifying ambulance station assignments and introducing additional ambulance relocations at the beginning of a shift. The main algorithm for the CH and a sub-function to introduce new ambulances into the system are shown in Figure 7-9 and Figure 7-10 respectively.

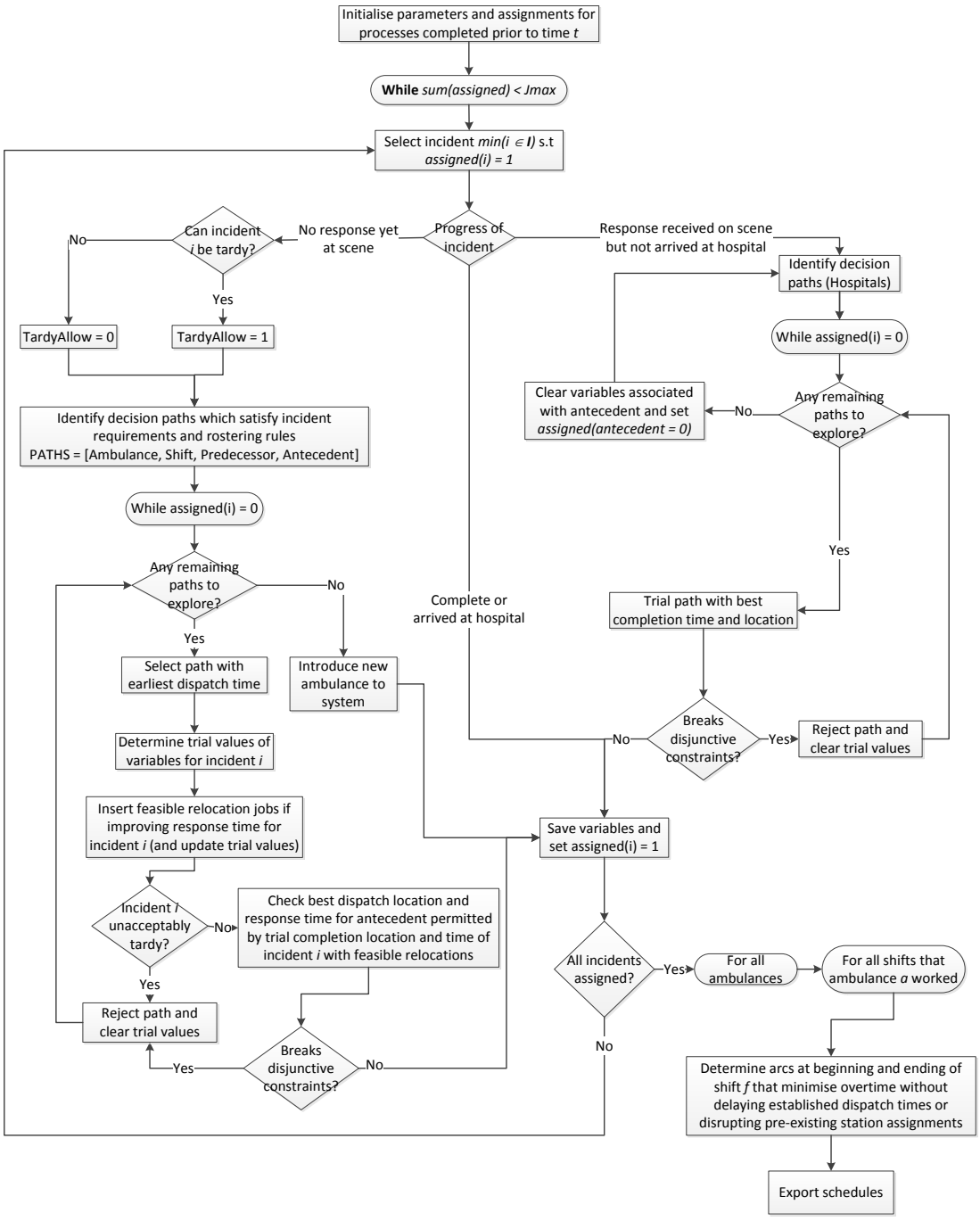


Figure 7-8 Process diagram for the CH for the dynamic model

CH for Dynamic Ambulance Scheduling Model

1: Initialise parameters

2: **For** $i = 1$ *to* I_{max}

3: **If** incident status is $I^1(t) = 1$ or $I^2(t) = 1$ (i.e not yet arrived at the scene)

4: $AllowTardy = \begin{cases} 1, & rand < TarRej(P_{ip}, Q_p) \\ 0, & otherwise \end{cases}$

5: Identify feasible assignment options

$AmbOps: \boldsymbol{A}' \subset \boldsymbol{A}(\boldsymbol{t})\ s.t.\ \xi_{i\boldsymbol{A}'}(t) = 1$

$ShiftOps: \boldsymbol{F}' \subset \boldsymbol{F}\ s.t.\ B_{\boldsymbol{F}'} < U_i(t)\ \&\ E_{\boldsymbol{F}'} \geq R_i(t)$

$AssignOps: \boldsymbol{x}' = [\boldsymbol{A}', f \in \boldsymbol{F}']$

6: **If** $\sum_{s \in \boldsymbol{S}} v_{a's f'}(t) = 0\ \forall a' \in \boldsymbol{A}', f' \in \boldsymbol{F}'$

Test whether assigning new shift $v_{a'f'}' = 1$ violates rostering rules

7: **If** $\sum_{f=f'-2}^{f'+2} \sum_{s \in \boldsymbol{S}} v_{a's f} > 0$ (min time off requirement)

Or If $\sum_{f \in \mathbb{F}\boldsymbol{w}} \sum_{s \in \boldsymbol{S}} v_{a's f'}(t) = 4, \forall w \in \boldsymbol{W}\ s.t.\ f' \in \mathbb{F}\boldsymbol{w}$ (max shifts p.w.)

Or If $f' \in \boldsymbol{G}$ & limit consecutive night shifts:

$$\text{any}\begin{pmatrix} \sum_{s \in \boldsymbol{S}} v_{a's(f'-6)}(t) + \sum_{s \in \boldsymbol{S}} v_{a's(f'-3)}(t) = 2 \\ \sum_{s \in \boldsymbol{S}} v_{a's(f'-3)}(t) + \sum_{s \in \boldsymbol{S}} v_{a's(f'+3)}(t) = 2 \\ \sum_{s \in \boldsymbol{S}} v_{a's(f'+3)}(t) + \sum_{s \in \boldsymbol{S}} v_{a's(f'+6)}(t) = 2 \end{pmatrix}$$

Or If $v_{a'f'}$ would prevent weekly RDO period occurring on previous week, current week or following week for *f'*

8: Remove $[a', f']$ from $\boldsymbol{x}'$

9: **End if**

10: **End if**

11: Determine all feasible predecessors and antecedents for each path in *AssignOps*

$$PreOps: \boldsymbol{J}' \subset \boldsymbol{J}\ s.t.\ \sum_{a \in \boldsymbol{A}', f \in \boldsymbol{F}'} \left(x_{\boldsymbol{J}'af}(t)\right) > 0\ \&\&\ \sum_{a \in \boldsymbol{A}'} c_{\boldsymbol{J}'a}(t) < \min\left(U_i(t), \sum_{a \in \boldsymbol{A}', f \in \boldsymbol{F}'} E_f \times x_{\boldsymbol{J}'af}(t)\right)$$

$$AnteOps: \boldsymbol{J}'' \subset \boldsymbol{J}\ s.t.\ \sum_{a \in \boldsymbol{A}', f \in \boldsymbol{F}'} \left(x_{\boldsymbol{J}''af}(t)\right) > 0\ \&\&\ \sum_{a \in \boldsymbol{A}'} d_{\boldsymbol{J}''a}(t) > R_i(t)$$

12: Build list of paths ordered by earliest ready to dispatch time (*d'*) where possible dispatch times are completion time of predecessor or starting time of shift for each allowable ambulance

$$PathOps1: [(a,f) \in \boldsymbol{x}',\ j \in \boldsymbol{J}'\ s.t\ x_{jaf}(t) = 1, \begin{pmatrix} j' \in \boldsymbol{J}''\ \ s.t. Z_{jj'}(t)=1 \\ 0\ iff\ \ \sum_{j' \in \boldsymbol{J}} z_{jj'af} = 0 \end{pmatrix}, d' = c_{ja}(t)]$$

$$PathOps2: [(a,f) \in \boldsymbol{x}', 0, j' \in \boldsymbol{J}''\ \ s.t. \sum_{j \in \boldsymbol{J}} z_{jj'af} = 0, d' = B_f]$$

PATH_OPS = sort({PathOps1; PathOps2}, d')

13: Set assigned = 0

CH for Dynamic Ambulance Scheduling Model cont'd

While assigned = 0
Trial next option in *PATH_OPS*
If no more options exist
Introduce new ambulance and set assigned = 1;
Else Determine trial values for all incident variables returning earliest response times, allow feasible relocations prior to dispatch if improves response time
If incident response time unacceptably tardy
Reject option and clear all trial values for variables
Else Trial new dispatch location for trial antecedent, allowing completion location or relocation for incident i
If *min(trial completion time for incident i + travel time to best dispatch location for trial antecedent + travel time from best dispatch location incident scene for antecedent) > response time of antecedent*
Then overlap exists between incident i and trial antecedent, reject option and clear trial values
Else accept all trial values for variables, including new dispatch location and time for antecedent, and set assigned =
End If
End If
End If
End While
ElseIf incident status is $I^3(t) = 1$ or $I^4(t) = 1$
(i.e arrived at the scene but not yet at hospital)
Identify antecedent j (if it exists) from $Z_{ij}(t)$
If hospital reassignment and/or relocation can improve solution
Update improved variables
End If
End If
End For
For each ambulance and shift updated this horizon
Assign appropriate return to station job with smallest overtime value
End For
Export results

Figure 7-9 Algorithm for the CH used to solve the dynamic ambulance scheduling model

Assign New Dynamic

Load stochastic parameters
Select *k* from $(k\,|i)$, depending on type of ambulance requested by incident *i*
Add new ambulance *a* of type *k* ($\boldsymbol{A_k} = \{\boldsymbol{A_k}, a\}$)
Assign earliest dispatch time (i.e. $d_{ia} = R_i$)
Find earliest shift options $f = \arg\left(\max_{f \in \boldsymbol{F}'}\left(B_f\right)\right)$ where $\boldsymbol{F}' \subset \boldsymbol{F}\ s.t\ B_{\boldsymbol{F}'} < R_i$
Select hospital $h = \arg(\min_{\mathrm{h} \in \mathbf{H}'}(\psi_{ih} + \zeta_{sh}))$ *where* $\boldsymbol{H}' \in \boldsymbol{H}$ s.t. $\gamma_{i\boldsymbol{H}'} > 0$
Identify completion location $c_l = \begin{cases} L_h,\ \gamma_{i\boldsymbol{H}'} > 0 \\ L_i,\ otherwise \end{cases}$
Set probability for each ambulance station $s \in \boldsymbol{S}$

$$P(s) = \begin{cases} \exp\left(\frac{-\theta_{is}}{U_i - R_i}\right),\ \theta_{is} \le D_i - R_i \\ 0,\ \theta_{is} > D_i - R_i \end{cases}$$

$$\bar{P}(s) = \frac{P(s)}{\sum_{s' \in \boldsymbol{S}} P(s')}$$

If $\sum_{s \in \boldsymbol{S}} \bar{P}(s) = 0$
Then select ambulance station with shortest response time

$$s = \arg\left(\min_{s' \in \boldsymbol{S}}(\theta_{is'})\right)$$

Else select random *s* from $\bar{P}(s)$
End If
Determine expected c'_{ia} & τ'_{af}
Probability of selecting relocation from ambulance station *s'* to *s* prior to dispatching ambulance to incident *i*

$$P(s') = \begin{cases} \max\left(0, c'_{ia} + \theta_{c_l s}\right), \text{if } \theta_{ss'} + \max\left(t, B_f\right) \le d_{ia} \\ \infty,\ \text{otherwise} \end{cases} \qquad \forall s' \in \boldsymbol{S}$$

$$\bar{P}(s') = \frac{\exp\left(P(s'')\right)}{\sum_{s'' \in \boldsymbol{S}} \exp\left(P(s'')\right)}$$

Select random *s'* from $\bar{P}(s)$
If $s' \neq s$
Then $s = s',\ x_{as'} = 1$ & insert relocation job
Update variables $(d_{ia}, r_{ia}, e_{ih}, g_{ih}, c_{ia}, \tau_{af})$ for incident *i* and ambulance *a*
End If
If $\tau_{af} > 0$ && $B_{f+1} + \min_{s \in \boldsymbol{S}} \theta_{is} \le \min(U_i, D_i + M \times AllowTardy)$
(i.e. incident *i* can be delayed until the next shift *f+1*
If $rand \le OverAccept$
Delay incident *i* and update variables
$s = \arg(\min_{s' \in \boldsymbol{S}}(\theta_{is'}))\,, f = f + 1, d_{ia} = B_f, \tau_{af} = 0\ etc.$
End If
End If

Figure 7-10 Algorithm for assigning new ambulances into the dynamic scheduling model

TS+CH solution algorithm

Initialise model parameters and generate initial incumbent solution x from CH algorithm
Store $x \rightarrow x^*$, $f(x) \rightarrow f^*(x)$, $x \rightarrow x^\dagger$, $f(x) \rightarrow f^\dagger(x)$, $TL \rightarrow \emptyset$, $i = 0$
while solve time < time limit
    Set $x = x^\dagger$, $f(x) = f^\dagger(x)$, Set $x^{\dagger\dagger} = x^\dagger$, $f^{\dagger\dagger}(x) = \infty$, $tabu = TL$
    while count < iteration limit for inner loop
        $x' = x$
        Calculate measure of benefit of swapping each incident j:
        $u(j) = g(delay, tardy, overtime, makespan)$,
        $j_1 = \arg\left(\max_{j \in \mathbf{J}} u(j)\right)$ & $j_2 = \arg\left(\min_{j \in \mathbf{J}\ s.t. j < j_1} u(j)\right)$
        Swap $x'(j_1) \leftrightarrow x'(j_2)$ and add $(j_1, j_2) \rightarrow tabu$
        $f'(x') = Rebuild\_CH(x')$ [rebuild solution with latest sequence]
        if $f'(x) < f^{\dagger\dagger}(x)$
            $x^{\dagger\dagger} = x'$, $f^{\dagger\dagger}(x) = f'(x)$ [update neighbourhood solution]
            $tabu' = (j_1, j_2)$
        end if
        if entire neighbourhood searched
            $count = \infty$
        else $count = count + 1$
        end if
    end while
    Set $x^\dagger = x^{\dagger\dagger}$, $f^\dagger(x) = f^{\dagger\dagger}(x)$
    $TL = [TL; tabu']$;
    if $size(TL)$ > tabu list limit
        $TL(1,:) = []$; [Remove oldest entry]
    end if
    if $f^\dagger(x) < f(x)$
        $x = x^\dagger$, $f(x) = f^\dagger(x)$ [update global solution]
    end if
end while
Return x and $f(x)$

Figure 7-11 Hybrid TS+CH heuristic to solve the dynamic model

7.3.4 Hybrid Tabu Search and Constructive Heuristic

The heuristic presented here is an extension of the TS+CH heuristic presented in Chapter 6. Three minor changes have been made. Firstly, the stopping condition is simplified to a single stopping condition based on solution time. The stopping conditions for the number of total iterations and number of iterations without improvement have been removed. Secondly, the limit on the number of swaps from within each neighbourhood has also been improved. The earlier version of the TS+CH heuristic selects exactly the designated number of swaps to sample in each neighbourhood. If this number exceeds the total number of unique swaps in a neighbourhood, the earliest entries are removed from the tabu list to allow the required number of swaps to be explored. With a rolling horizon solution approach, small horizons are more likely to lead to small problem sizes where the entire neighbourhood could easily be explored. The updated TS+CH heuristic includes a break in the code to allow the next neighbourhood to explored as soon as all possibilities in the previous one are searched, regardless of whether or not the desired limit of incident swaps is covered. Finally, the method of storing data from the TS+CH solver has been streamlined so that it is easy to call with the rolling horizon. The new algorithm for the TS+CH, as utilised to solve the dynamic ambulance scheduling and shift scheduling model, is shown in Figure 7-11.

7.3.5 Ant Colony Optimisation

Ant Colony Optimisation, as described in Chapter 4, is an iterative heuristic which places pheromone on the disjunctive arcs forming feasible solutions. Each iteration sends out a number of ants to traverse the solution space. These ants are influenced by pheromone laid down by ants which previously explored the solution space. The amount of pheromone placed on an arc depends on the quality of the solution found through selecting that arc. Solutions are expected to converge as pheromone builds on the arcs leading to good solutions and evaporates on the arcs leading to poorer results.

The application of ACO to the dynamic ambulance scheduling and shift scheduling model places pheromone on arcs representing assignment decisions. These arcs are illustrated in Figure 7-12, representing sequencing decisions, assignment of preceding incidents (disjunctions), ambulances, shifts, stations and

hospitals. This creates an extremely large number of paths for ants in the heuristic to traverse and a large number of nodes where pheromone must be deposited. It is expected that tuning the parameters in this model can lead to good solutions but the time to reach convergence may be large.

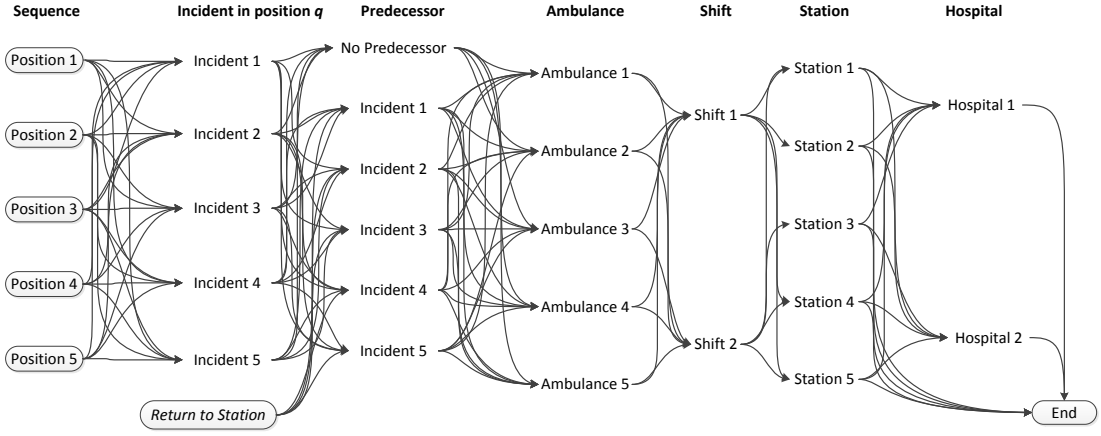


Figure 7-12 Decision arcs for Ant Colony Optimisation heuristic

The process illustrated in Figure 7-13 involves selecting paths to take at each junction based on the amount of pheromone specific to that type of decision. Pheromone directly influences the following decisions:

- determining the sequence in which incidents are assigned;
- assigning predecessors and/or antecedents to incidents;
- assigning ambulances to ambulance stations;
- assigning incidents to hospitals;
-

and indirectly influences the following decisions:

- assigning incidents to ambulances (through predecessors/antecedents);
- assigning incidents to shifts (through predecessors/antecedents);
- assigning ambulances to shifts (through assigning incidents to ambulances).

The ACO process, may be described as follows. Ant Colony Optimisation is initialised for each horizon within the rolling horizon approach used to solve the dynamic model. The ACO process continues to run until a stopping condition, in this case a time limit, is met. Each iteration within the ACO algorithm explores *n* paths,

where *n* is the number of ants defined in the parameters. The first junction in each path selects the next incident to be sequenced, from all unassigned incidents. This decision is influenced by previous solutions through the application of pheromone. All feasible ambulance and shift assignments are then identified, with all other incidents already assigned onto those ambulances determining possible dispatch times within the response time window for the current incident. While a feasible assignment remains to be found for the current incident, an ambulance assignment and dispatch time will be selected from the options by using a probability determined by pheromone from previous solutions, combined with an independent indicator of the quality of the option (in this case the expected response time and overtime). If the option tested results in a feasible solution, including a hospital selection that has no overlap with successive incidents already scheduled on the same ambulance, the assignment for the incident is fixed, and the process moves on to selecting the next incident to assign. If none of the existing options are feasible, a new ambulance is introduced into the system. The ambulance type, ambulance station and shift assigned to this ambulance depend on the requirements of the incident and are also influenced by pheromone on the decision arcs. Pheromone is only updated after each iteration of *N* ants, however, the global solution is updated everytime an improving solution is found.

The pheromone for each type of decision is able to have different parameters expressing its influence, application and evaporation, but all pheromone data updates at the same time. Each time a new solution is sought, it is rebuilt from the existing pool of ambulances and pheromone. The ACO rebuilding heuristic is based on the same logic as the CH but utilises probabilities obtained from the ACO methodology to select paths where the CH assigns best options on a FCFS basis. The ACO algorithm for constructing solutions is shown in Figure 7-15, with a sub-function for assigning new ambulances into the system in Figure 7-16.

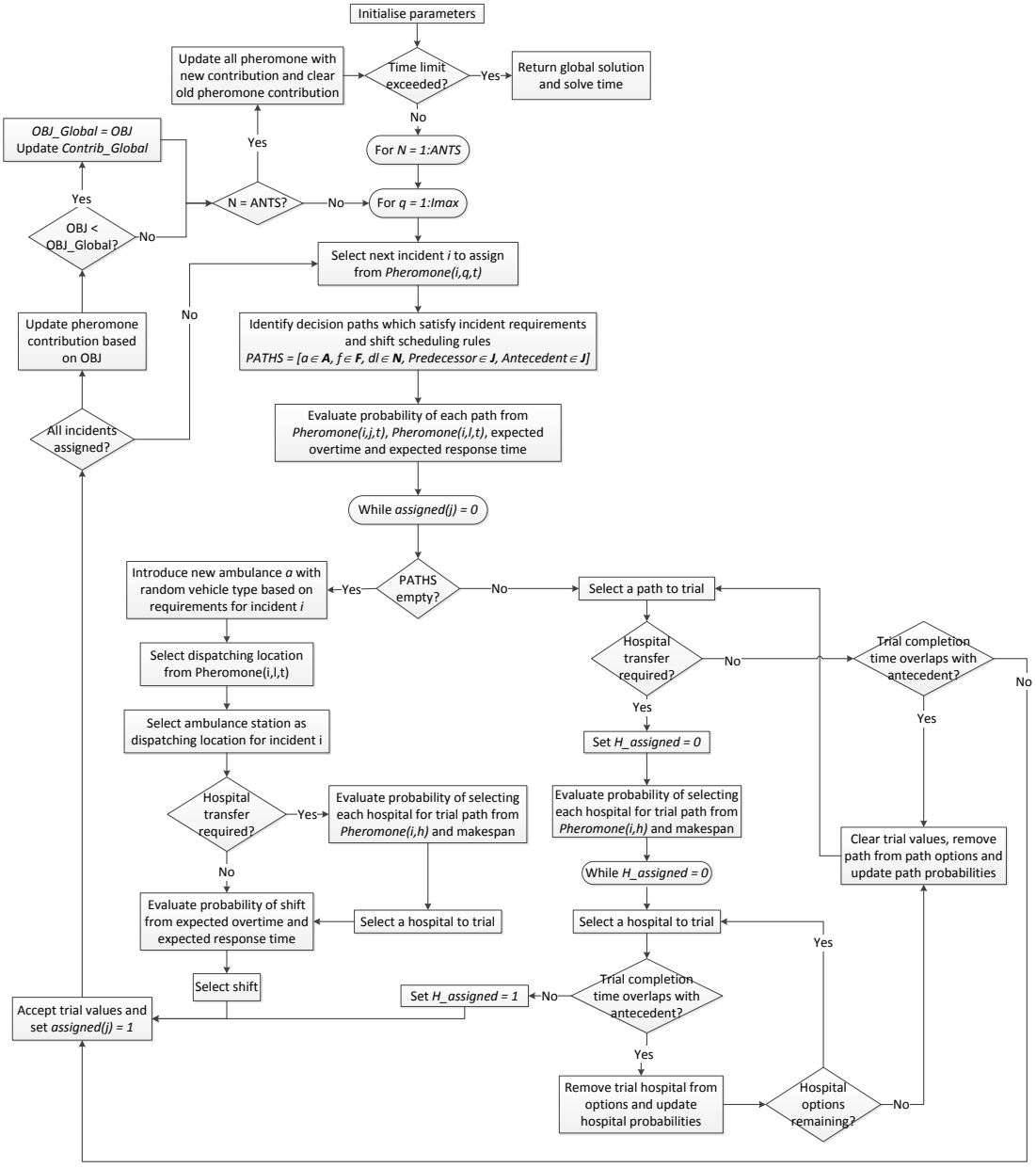


Figure 7-13 Process diagram for Ant Colony Optimisation

ACO algorithm

1: Initialise parameters

2: **For** $q = 1$ *to* I_{max}

3: Select next incident i to be assigned in sequence q given:

$$Prob(i|q) = \frac{\big(1 - assigned(i)\big) * PH(i,q) * k(i)}{\sum_{i' \in \boldsymbol{I}} \Big(\big(1 - assigned(i')\big) * PH(i',q) * k(i)\Big)}$$

$$\text{where } k(i) = \begin{cases} \left(\frac{1}{DueTimeTardy\ (i)}\right)^{\beta_{seq}}, & DueTimeTardy(i) < \infty \\ 1, & DueTimeTardy(i) = \infty \end{cases},$$

and R_{seq} determines if selection is deterministic or random.

4: **If** incident status is $I^1(t) = 1$ or $I^2(t) = 1$ (i.e not yet arrived at the scene)

5: $$AllowTardy = \begin{cases} 1, & rand < TarRej(P_{ip}, Q_p) \\ 0, & otherwise \end{cases}$$

6: Identify feasible assignment options

$AmbOps: \boldsymbol{A}' \subset \boldsymbol{A}(\boldsymbol{t})\ s.t. \xi_{i\boldsymbol{A}'}(t) = 1$

$ShiftOps: \boldsymbol{F}' \subset \boldsymbol{F}\ s.t. B_{\boldsymbol{F}'} < U_i(t)\ \&\ E_{\boldsymbol{F}'} \geq R_i(t)$

$AssignOps: \boldsymbol{x}' = [\boldsymbol{A}', f \in \boldsymbol{F}']$

7: **If** $\sum_{s \in \boldsymbol{S}} v_{a's f'}(t) = 0\ \forall a' \in \boldsymbol{A}', f' \in \boldsymbol{F}'$

Test whether assigning new shift $v_{a'f'} = 1$ violates rostering rules

8: **If** $\sum_{f=f'-2}^{f'+2} \sum_{s \in \boldsymbol{S}} v_{a's f} > 0$ (min time off requirement)

9: **Or If** $\sum_{f \in \mathbb{F}\boldsymbol{w}} \sum_{s \in \boldsymbol{S}} v_{a's f'}(t) = 4, \forall\, w \in \boldsymbol{W}\ s.t. f' \in \mathbb{F}\boldsymbol{w}$

10: **Or If** $f' \in \boldsymbol{G}$ & limit on consecutive night shifts by

$$\text{any}\begin{pmatrix} \sum_{s \in \boldsymbol{S}} v_{a's(f'-6)}(t) + \sum_{s \in \boldsymbol{S}} v_{a's(f'-3)}(t) = 2 \\ \sum_{s \in \boldsymbol{S}} v_{a's(f'-3)}(t) + \sum_{s \in \boldsymbol{S}} v_{a's(f'+3)}(t) = 2 \\ \sum_{s \in \boldsymbol{S}} v_{a's(f'+3)}(t) + \sum_{s \in \boldsymbol{S}} v_{a's(f'+6)}(t) = 2 \end{pmatrix}$$

11: **Or If** $v_{a'f'}$ would prevent weekly RDO period occurring on previous week, current week or following week for f'

12: Remove $[a', f']$ from $\boldsymbol{x}'$

13: **End if**

14: **End if**

15: Determine all feasible predecessors and antecedents for each path in *AssignOps*

$$PreOps: \boldsymbol{J}' \subset \boldsymbol{J}\ s.t. \sum_{a \in \boldsymbol{A}', f \in \boldsymbol{F}'} \Big(x_{\boldsymbol{J}'af}(t)\Big) > 0\ \&\&\ \sum_{a \in \boldsymbol{A}'} c_{\boldsymbol{J}'a}(t) < \min\left(U_i(t), \sum_{a \in \boldsymbol{A}', f \in \boldsymbol{F}'} E_f \times x_{\boldsymbol{J}'af}(t)\right)$$

$$AnteOps: \boldsymbol{J}'' \subset \boldsymbol{J}\ s.t. \sum_{a \in \boldsymbol{A}', f \in \boldsymbol{F}'} \Big(x_{\boldsymbol{J}''af}(t)\Big) > 0\ \&\&\ \sum_{a \in \boldsymbol{A}'} d_{\boldsymbol{J}''a}(t) > R_i(t)$$

16: $r' = \max\big(t, R_i, B_f, (c_{ante}, a + reloc\_time)\big) + required\ travel\ tme$

ACO algorithm cont'd

Build list of paths ordered by earliest feasible response time (r')

$PathOps1: [(a,f) \in \boldsymbol{x}',\ j \in \boldsymbol{J}'\ s.t\ x_{jaf}(t) = 1, \begin{pmatrix} j' \in \boldsymbol{J}''\ s.t. Z_{jj'}(t)=1 \\ 0\ iff\ \sum_{j' \in \boldsymbol{J}} z_{jj'af}=0 \end{pmatrix}, r']$

$PathOps2: [(a,f) \in \boldsymbol{x}', 0, j' \in \boldsymbol{J}''\ s.t. \sum_{j \in \boldsymbol{J}} z_{jj'af} = 0, r']$

PATH_OPS = sort({PathOps1; PathOps2}, r')

Set assigned = 0

If no feasible assignment options exist

Run *ACO_newassign* to introduce new ambulance for incident *i* and set assigned = 1;

Else Calculate probabilities for next option to trial

For a = 1:length(PATH_OPS)

If $\sum_{s \in \boldsymbol{S}, f \in \boldsymbol{F}} v\_asf(t) = 0$

Select ambulance station to trial for this options

$$Prob(a \to i \mid a \to s) = \frac{PH(i,s) * k(i,s)}{\sum_{s' \in \boldsymbol{S}} PH(i,s') * k(i.s')}$$

where $k(i,s) = \mu_{l_s,l_i}(t)^{\beta_{station}}$.

Use R_{JS} to determine if station selection is deterministic or random.

End If

Calculate trial values and expected overtime

$$PHOptions = PH(PreOps(a), i) + PH(i, AnteOps(a) + \sum_{j \in \boldsymbol{J}} PH(j, PreOps(a)) * \sum_{a' \in \boldsymbol{A}, f \in \boldsymbol{F}} z_{jPreOps\ (a)a'f} + \sum_{j \in \boldsymbol{J}} PH(AnteOps(a), j) * \sum_{a' \in \boldsymbol{A}, f \in \boldsymbol{F}} z_{AnteOps\ (a)ja'f}$$

$$NonPHOptions(a) = \left(\frac{1}{(expected\ \ overtime\ (a)}\right)^{\beta_{ot}} \times \left(\frac{1}{(r'(a) - R_i)}\right)^{\beta_{response}}$$

End For

$$Prob(a \to i) = \frac{PHOptions\ (a) * NonPHOptions\ (a)}{\sum_{a \in PATHOPS}\ HOptions\ (a) * NonPHOptions\ (a)}$$

While assigned = 0

Use R_{JMF} to determine if selection of path options is deterministic or not

Select option to trial and determine values across all hospital paths

If incident response time unacceptably tardy or dispatch after shift end

Reject trial, remove from options, clear all trial values

Else Trial hospital selection with possible new dispatch location for trial antecedent

End If

Calculate probability for assigning each hospital

$$Prob(h|i) = \frac{PH(i,h) * k(i,h)}{\sum_{(h \in \boldsymbol{H})} PH(i,h) * k(i,h)}$$

where $k(i,h) = \left(\frac{1}{makespan\ (i,h)}\right)^{\beta_{hospital}}$

ACO algorithm cont'd

Use R_H to determine if selection of hospital options is deterministic
Set Hassigned = 0
While Hassigned = 0
    Check whether trial option obeys all constraints
    If *min(trial completion time for incident i + travel time to best dispatch location for trial antecedent + travel time from best dispatch location incident scene for antecedent) > response time of antecedent*
    Then overlap exists between incident i and trial antecedent,
        Reject trial hospital assignment and remove from options
            If no more hospital paths to trial
                Update trial values with antecedent as new predecessor
                Set Hassigned = 1
            End If
    Else Accept all trial values for variables, including new dispatch location and time for antecedent,
        Set assigned = 1
    End If
End While
End While
End If
Else If incident status is $I^3(t) = 1$ or $I^4(t) = 1$
Calculate probability for assigning each hospital

$$Prob(h|i) = \frac{PH(i,h) * k(i,h)}{\sum_{(h \in \boldsymbol{H})} PH(i,h) * k(i,h)}$$

where $k(i,h) = \left(\frac{1}{makespan\ (i,h)}\right)^{\beta_{hospital}}$

Use R_H to determine if selection of hospital options is deterministic
Set Hassigned = 0
While Hassigned = 0
    Select hospital assignment to trial
    Trial new dispatch location for antecedent (if any), allowing completion location for incident i or feasible relocation
    If *min(trial completion time for incident i + travel time to best dispatch location for trial antecedent + travel time from best dispatch location incident scene for antecedent) > response time of antecedent*
    Then Overlap exists between incident i and trial antecedent,
        Reject trial hospital options and clear trial values
    Else Accept all trial values, including new dispatch location and time for antecedent,
        Set Hassigned = 1
    End If

ACO algorithm cont'd

70: **If** all options rejected
71: Select hospital returning smallest completion time
72: Assign variables for incident *i* and set Hassigned = 1
73: Reassign antecedent to another ambulance:
74: **End If**
75: **End While**
76: **End If**
77: **For** each ambulance and shift updated this horizon
78: Assign appropriate return to station job with smallest overtime value
79: **End For**

Figure 7-14 Ant Colony Optimisation for the dynamic model

Assign New ACO

1: Initialise parameters
2: Assign new ambulance *a* to incident *i* with $d_{ia}(t) = R_i(t)$
3: Randomly select ambulance type $k \in \boldsymbol{K}$ according to $Prob(k, \kappa_i(t))$
4: Select station from $Prob(s|i) = \frac{PH(i,s)*k(i,s)}{\sum_{s \in \boldsymbol{S}} PH(i',s)*k(i',s)}$
where $k(i,s) = \mu_{l_s,l_i}(t)^{B_{station}}$
5: Use R_{JS} to determine if station selection is deterministic or random
6: Determine trial values for all variables across all hospital paths
7: Calculate probability for assigning each hospital $Prob(h|i) = \frac{PH(i,h)*k(i,h)}{\sum_{(h \in \boldsymbol{H})} PH(i,h)*k(i,h)}$
where $k(i,h) = \left(\frac{1}{makespan\ (i,h)}\right)^{\beta_{hospital}}$
8: Use R_H to determine if selection of hospital options is deterministic
9: Identify all feasible shifts $\boldsymbol{F'}$ for incident *i*
10: Calculate probability of selecting each shift $Prob(f|i) = \frac{k(f)}{\sum_{f \in \boldsymbol{F'}} k(f)}$ where

$$k(f) = \max\left(0, T_i(t) - max\left(R_i(t), B_f(t)\right) + \mu_{l_s,l_i}(t)\right) * 120 + \max\left(0, c_{ia}(t) + \mu_{\sum_{n \in \boldsymbol{N}} \theta_{in}(t), l_s)} - E_f(t)\right),$$

11: Randomly select shift
12: Update and save all variables for incident *i*

Figure 7-15 ACO algorithm for introducing new ambulances into the system for the dynamic ambulance scheduling model

7.3.6 Hybrid Ant Colony Optimisation and Constructive Heuristic

This hybrid heuristic uses Ant Colony Optimisation to determine the sequence in which incidents are assigned to ambulances, but rebuilds the schedules using the constructive heuristic. It is anticipated that this will improve solution times when compared with the TS+CH heuristic, as it can swap the position of multiple incidents

for each iteration at the beginning of the solution period, and then perform fewer swaps as the pheromone is established across arcs. It requires much less memory than the ACO heuristic and is expected to converge faster. The decision arcs for the sequence in which incidents are considered are illustrated in Figure 7-16. This shows that any incident may be placed in any position, although each position can only be assigned a single incident each time.

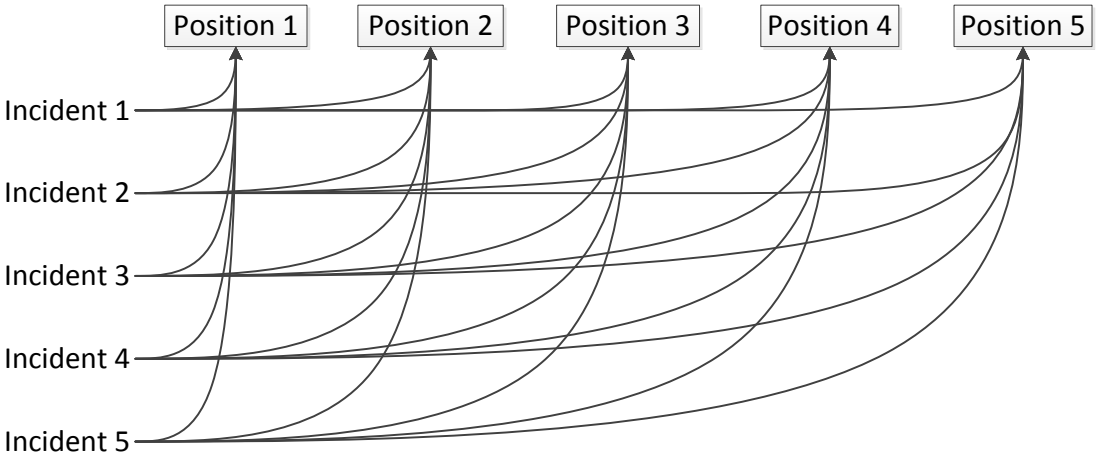


Figure 7-16 Sequencing arcs for ACO+CH hybrid heuristic

As there are a large number of parameters which may be tuned for this model, multiple sets of parameter values are investigated. The first version test (ACO +CH.1) uses the largest number of ants in the heuristic and, as such, is able to explore more paths every iteration but will update pheromone less frequently. The second version (ACO+CH.2) uses almost all of the same parameters as ACO+CH.1 with the exception of the number of ants, which has been reduced. It is expected that, for larger problems, updating the pheromone more frequently will lead to a faster convergence and better solutions in the same amount of time. The third version (ACO+CH.3) uses the same number of ants as the ACO+CH.2, but increases parameter R so that path selections are more deterministic, and reduces α to slow the evaporation rate of pheromone. These changes are expected to allow good solutions found early to be remembered for longer and have more influence on path selection. For scenarios where a solution is needed quickly, early good solutions are of a great deal of interest. A summary of the parameters used for the three versions of the ACO+CH heuristic is shown in Table 7-1.

Table 7-1 Tuning parameters for ACO+CH hybrid heuristic

	ACO+CH.1	**ACO+CH.2**	**ACO+CH.3**
ANTS	*50*	*20*	*20*
α	*0.5*	*0.5*	*0.2*
R_{seq}	*0.5*	*0.5*	*0.8*

The process for the hybrid ACO+CH is illustrated in Figure 7-17. The ACO+CH process uses pheromone for ordering all incidents but builds schedules using the Constructive Heuristic. The ACO component acts as an outer heuristic, selecting incidents to place in each position for scheduling until all positions are filled with unique incidents. This information affects the parameters passed on to the CH. Schedules and objective function values from the CH are then used to update the pheromone used in the ACO section of the hybrid heuristic. The stopping condition for the ACO+CH, as with the other heuristics presented in this chapter, is a limit on the time spent trying to find a solution. The algorithm shown in Figure 7-17 presents the hybrid heuristic in more detail.

When sequencing incidents, positions are filled in ascending order based on the probability that each incident occupies that exact position. The benefit of this approach is the ability to explore a large number of incident moves each iteration so that the sequences far from the original may be reached quickly. A flaw of this approach is that an incident i which performs well in position q may also perform well in the positions neighbouring q (i.e. q^*), but the pheromone $PH(i,q)$ does not influence the probability of assigning incident i to any positions in q^*, and nearby improving solutions may never actually be explored. Possible methods to correct this flaw should be explored in future work. Proposed amendments to the ACO+CH include reducing the set of incidents which may be re-sequenced from all positions to smaller sections of neighbouring positions. For example, the ACO currently allows incidents to be swapped in all positions from 1 to I_{max}, but it is possible to only allow incidents within the first third of positions to be moved to another position within the first third of options. Varying the size of the section wherein incidents may be moved is an extension of this idea. Alternatively, the ACO+CH could be hybridised further with the inclusion of TS to refine incident sequencing after the ACO+CH is complete.

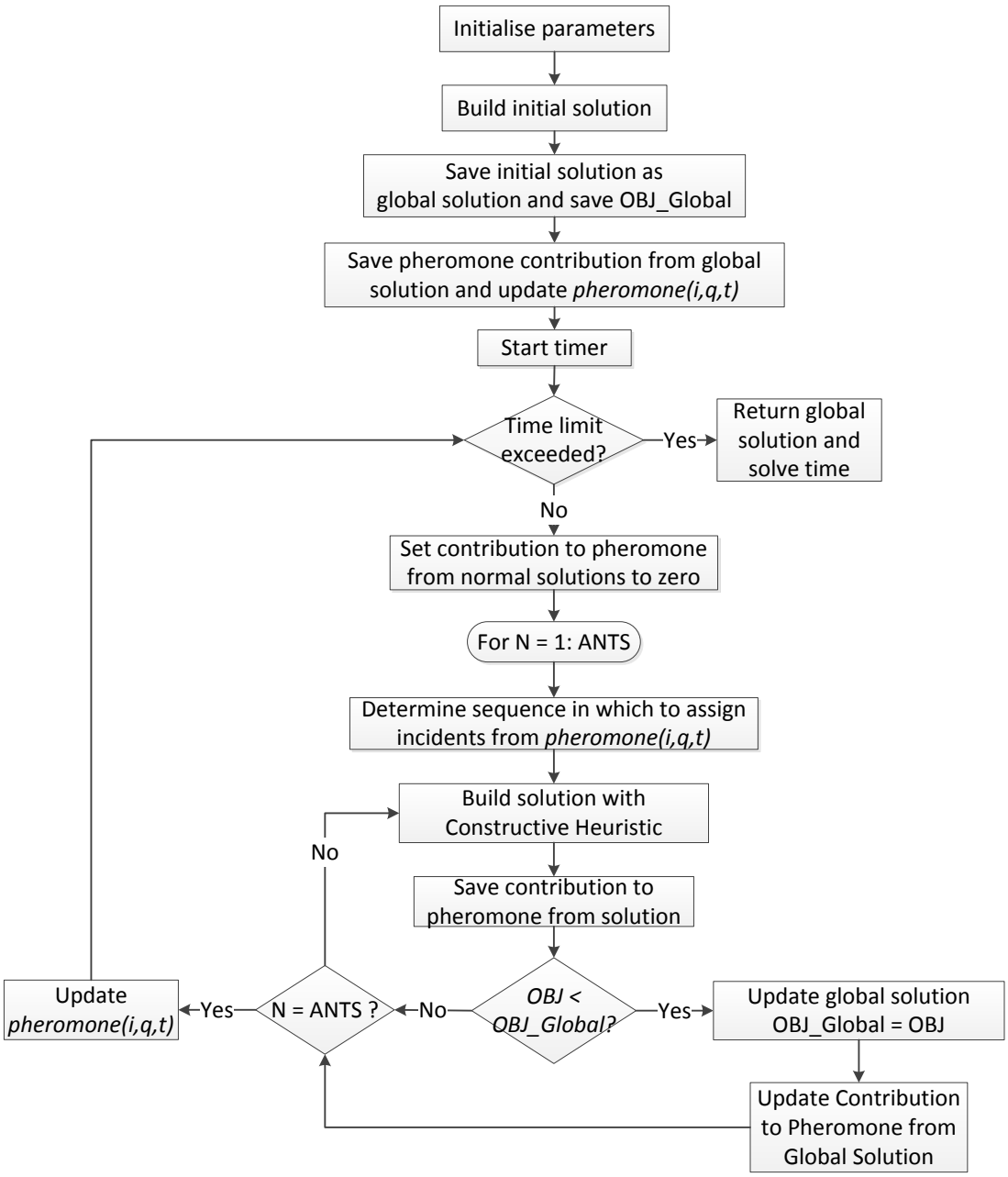


Figure 7-17 Process diagram for hybrid ACO+CH heuristic for the dynamic model

Hybrid ACO+CH algorithm

Initialise parameters
Let *O*= initial order of incidents and define incident input as *I(O)*
Set initial solution: $x = CH(I(O))$
Save initial solution as best solution: $x^{BEST} = x$, $f^{BEST}(x) = f(x)$ and best order of incidents: $O^{BEST} = O$
Update pheromone $PH(i,q) = \min\{\tau_0, (1-\alpha)PH(i,q) + \alpha * U\frac{C_{iq}^{BEST}}{f(x^{BEST})\hat{}\delta}$
where $C_{iq}^{BEST} = \begin{cases} 1, & O(i) = q \\ 0, & otherwise \end{cases}$ (i.e. identity matrix of size (I_{max}, I_{max}))
While *solve time < solve time limit*
    Set $\Delta PH(i,q) = 0 \; \forall \; i \in \boldsymbol{I}, q = 1 \; to \; I_{max}$
    For *N = 1:ANTS*
        Set $assigned(i) = 0$ & $O^*(i) = 0 \;\; \forall i \in \boldsymbol{I}$
        For $q = 1 \; to \; I_{max}$
            Define $\boldsymbol{I}' \subset \boldsymbol{I}$ s.t. $T_{\boldsymbol{I}'} = \infty$ (i.e. relocation type jobs from previous horizon/s reinstated as incidents this horizon) and $\boldsymbol{I}'' = \boldsymbol{I} \,/\boldsymbol{I}'$
            Define probabilities
            $prob_a(i) = \big(1 - assigned(i)\big)PH(i,q) \; \forall i \in \boldsymbol{I}'$ and
            $$prob_b(i) = \big(1 - assigned(i)\big)PH(i,q)\left(\frac{1}{T_i}\right)^{\beta} \quad \forall \; i \in \boldsymbol{I}''$$
            Probability of selecting incident *i* for position *q*:
            $$Prob(i,q) = \frac{\begin{Bmatrix} prob_a(i) \; if \; i \in \boldsymbol{I}' \\ prob_b(i) if \; i \in \boldsymbol{I}'' \end{Bmatrix}}{\left(\sum_{i' \in \boldsymbol{I}'} prob_a(i') + \sum_{i' \in \boldsymbol{I}''} prob_b(i')\right)}$$
            If $rand \leq R_{seq}$
                $i = \arg\max_{j \in \mathbf{I}} Prob(j,q)$
            Else Randomly select *i* from *Prob(i,q)*
            End If
            Set $O(i) = q$, $assigned(i) = 1$ and $C_{iq} = 1$
        End For
        Define $I(O)$
        Obtain solution $x = CH(I(O))$
        $\Delta PH(i,q) = \Delta PH(i,q) + C_{iq}/\big(f(x)\big)^{\delta} \;\; \forall i \in \boldsymbol{I}, q = 1 \; to \; I_{max}$
        If $f(x) < f^{BEST}(x)$
            Save $x \rightarrow x^{BEST}$, $f(x) \rightarrow f^{BEST}(x), O \rightarrow O^{BEST}, C_{iq} \rightarrow C_{iq}^{BEST}$
        End If
    End For
    Update pheromone
$$PH(i,q) = \max\left\{PH_{min}, \min\{\tau_0, (1-\alpha)PH(i,q) + \alpha * U\left(\Delta PH(i,q) + \frac{C_{iq}^{BEST}}{f(x^{BEST})^{\delta}}\right)\right\}$$
End While
Return x^{BEST} , $f^{BEST}(x)$, and O^{BEST}

Figure 7-18 Algorithm for the ACO+CH heuristic for the dynamic ambulance model

7.4 RESULTS AND DISCUSSION

In this section, the performance of the heuristic is analysed and then the best performing solution methods used to solve the dynamic model for one week.

7.4.1 Quality of Heuristics

The quality of the heuristic solutions is verified through comparing heuristic results with results from a reduced problem able to be solved exactly within CPLEX.

7.4.1.1 Reduced problem

The reduced problem has the following characteristics:

- Five incidents
- One shift
- One period of time beginning at $t = 300$ minutes and ending two hours later at $t = 420$
- Five ambulance stations and two hospitals
- Two Type I ambulances, 1 Type II ambulance and 0 Type III ambulances
- Ambulance 1, with vehicle type I, pre-assigned to ambulance station 5
- Eleven potential return-to-station jobs
- Eighteen potential relocation jobs (four jobs directed to station 1, five jobs directed to station 2, two jobs directed to station 3, four jobs directed to station 4 and three jobs directed to station 5).

The aim of reducing the problem is to limit the number of jobs, without excluding the optimal solution, so that it can be solved in reasonable time by an exact MIP solver. Five incidents were selected during a period of non-peak demand. This is sufficient to ensure multiple ambulances will be required and that multiple jobs on a single ambulance are a possibility. Appropriate selection of the initial incident ensures that a shift boundary is investigated, so that overtime appears in the solution. Only three ambulances were selected as input for the reduced model, despite the potential to allocate up to five ambulances (one per incident). Pre-analysis of the problem solved with the heuristics showed that a good solution existed with two type I ambulances and one type II ambulance, and no improving solution that reduces the number or type of ambulances is possible. Limiting

ambulances limits the number of return-to-station jobs. This can be further reduced by noting that ambulance station five is the closest ambulance station to hospital two, to which the final incident must travel. This incident, due to release time and due dates, will incur overtime. It is already known that a feasible solution exists with this allocation from pre-solving the model with the heuristics. Selecting in advance the station which will reduce overtime will form part of the optimal solution. The number of potential relocation jobs is also reduced as much as possible, without risking the loss of the optimal solution. By assuming that relocations will only occur prior to incident (that is, relocations will not occur prior to return-to-station jobs or other relocations), it is possible to state that there will be a maximum of five relocation jobs for each destination. Relocation jobs are also removed if it is clear from travel times between locations that they will never form part of an optimal solution.

Solving the reduced problem in CPLEX returned an exact solution with the objective function value of 6.4732 Weighted Ambulances Hours, found after 25 hours. This is compared to the results from each of the heuristics for 10 tests, using the same five incidents and relaxing the restrictions on fleet size, vehicle types, station assignment and possible relocations. Table 7-2 shows the results. The hybrid heuristics are able to reach the optimal solution, with two of the three hybrid ACO+CH heuristics showing lower variability in the solution for the small size problem. The ACO+CH.1 heuristic actually performs the best for the small size problem, although it is shown in the next section that it is less effective with larger size problems.

Table 7-2 Best and average solutions for a test model with five incidents

Solution method	OBJ_{best} *(WAH)*	OBJ_{avg} *(WAH)*	*Var(OBJ)*	*average cputime (secs)*
MIP	**6.4732**	-	-	90920.88
CH	6.5674	7.4680	6.0443E-01	2.42
TS+CH	**6.4732**	6.4817	7.2067E-05	1280.76
ACO	6.4778	6.5520	8.2649E-04	1000.77
ACO+CH.1	**6.4732**	6.4735	3.7997E-07	1000.12
ACO+CH.2	**6.4732**	6.4741	3.6620E-06	1000.08
ACO+CH.3	**6.4732**	6.4785	1.062E-04	1000.16

7.4.1.2 Small sample problems

The quality of heuristics is further investigated for larger problem sizes which are unable to be solved exactly. A problem size of 20 incidents is chosen with average time between incidents of approximately two hours and maximum time between incidents of less than the duration of one shift. This allows peak and off-peak times to be investigated. A secondary problem size of 165 incidents is also selected to guarantee that every scenario compared will have at least one shift ending during the time interval covered. Scenarios are created by selecting random starting incidents and selected consecutive incidents until the problem size is reached. The characteristics of the scenarios are shown in Table 7-3 and results for 30 scenarios of each problem size are shown in Table 7-4 and Table 7-5.

Table 7-3 Characteristics of scenarios used to test heuristics for the dynamic model

# of incidents	**Time Interval (hours)**					
	Average	Min	10^{th} Percentile	50^{th} Percentile	90^{th} Percentile	Max
20	2.35	0.60	0.93	1.80	4.78	8.87
165	20.74	10.12	12.93	20.73	28.47	32.27

Table 7-4 Analysis of objective values from heuristics for 20 incidents across 30 scenarios in the dynamic model

Solution Approach	Average Objective (WAH)	Standard deviation	10^{th} Percentile Objective (WAH)	50^{th} Percentile Objective (WAH)	90^{th} Percentile Objective (WAH)	Average CPU runtime (secs)	Standard deviation
CH	15.15	3.79	9.79	15.77	19.60	3.81	1.19
TS+CH	14.75	2.42	11.95	15.00	17.55	1000.49	0.29
ACO.1	15.76	2.74	11.95	15.50	19.91	1001.37	4.39
ACO+CH.1	12.46	3.14	9.90	**12.38**	16.83	1000.35	0.16
ACO+CH.2	**12.34**	2.22	**9.49**	12.50	**15.50**	1000.35	0.23
ACO+CH.3	12.70	2.28	10.06	13.00	**15.50**	1000.33	0.22

Table 7-5 Analysis of objective values from heuristics for 165 incidents across 30 scenarios in the dynamic model

Solution Approach	Average Objective (WAH)	Standard deviation	10^{th} Percentile Objective (WAH)	50^{th} Percentile Objective (WAH)	90^{th} Percentile Objective (WAH)	Average CPU runtime (secs)	Standard deviation
CH	65.12	13.41	48.41	65.92	78.58	15.46	0.92
TS+CH	**55.38**	12.12	41.67	56.00	**71.02**	1004.24	1.86
ACO.1	60.52	15.14	44.42	**55.39**	78.53	1002.04	1.21
ACO+CH.1	62.94	11.89	44.47	67.31	75.39	1007.19	3.70
ACO+CH.2	61.46	12.71	44.74	62.85	76.98	1006.12	4.24
ACO+CH.3	57.55	11.23	**40.55**	58.48	71.65	1008.93	5.54

The results show that for small problem sizes, the hybrid ACO+CH heuristics outperform the other heuristics. The versions with parameters for a more random incident position selection are able to find better solutions (ACO+CH.1 and ACO+CH.2), possibly as a result of their ability to explore a greater area of the search space. The CH alone is not very effective but is fast to solve. The ACO heuristic provides the poorest solutions for 20 incidents but shows improvement for the problem with more incidents, although the average and best solutions for 165 incidents from the ACO heuristic are still outperformed by the ACO+CH.3 and TS+CH heuristics. The larger problem (165 incidents) has the best solutions from the hybrid TS+CH. The hybrid ACO+CH algorithms show better solutions, with parameters allowing more frequent updating of the pheromone (ACO+CH.2 and ACO+CH.3 with fewer ants) and again with parameters for a more deterministic approach (i.e. ACO+CH.3). From these results, the hybrid heuristics TS+CH and ACO+CH are most promising to investigate the model for a full week of incidents. The parameters from ACO+CH.3 are expected be of more benefit for the longest horizons where the ability to make multiple swaps each iteration would allow the solution space to be explored faster, and this heuristic is selected to solve for weekly results. Further work is possible to extend the heuristic to vary the parameters, depending on the problem size.

Investigating different scenarios of the same problem size allows investigation of solutions at times of peak and non-peak demand. Looking at problem sizes of 20 incidents and dividing the scenarios into peak demand (100 minutes or less between arrival of first and last incident) and off-peak (more than 100 minutes between arrival of first and last incident), it is possible to see that scenarios during peak demand result in higher objective function values; that is, periods of peak demand have a requirement for a higher number of weighted ambulance hours. This is expected, as more ambulances will be required simultaneously. This pattern was observed for all heuristics and is illustrated for the ACO+CH.3 heuristic in Figure 7-19, where the OFV for 100 scenarios is plotted against the time interval between the arrival of the first and last incident. The figure also shows an upper bound on the OFV of 15.5.

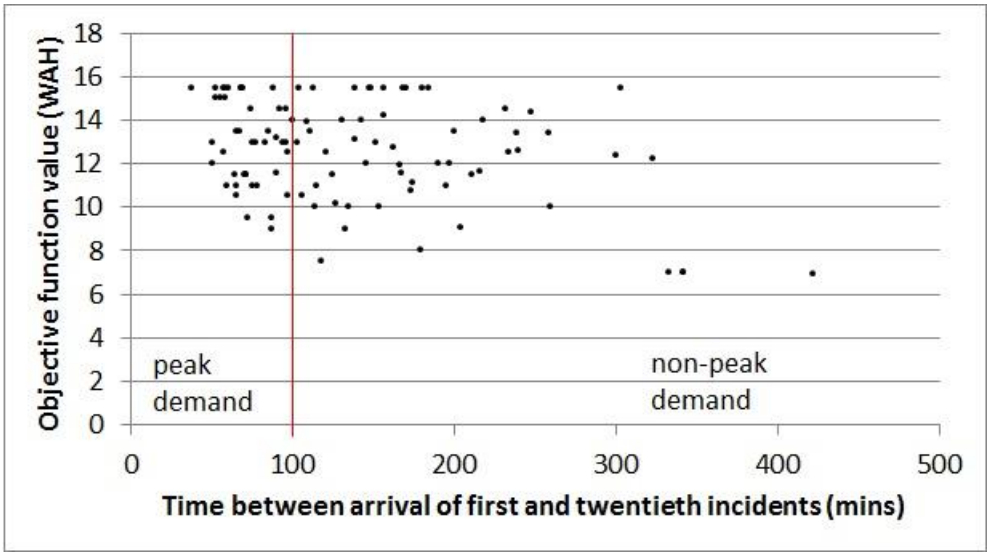


Figure 7-19 Analysis of the Objective Function Value for scenarios with a problem size of 20 incidents for the dynamic model

The moving averages of solutions from the TS+CH and ACO+CH.3 heuristics are investigated for specific scenarios of 20 and 165 incidents. Figure 7-20a shows the moving average for the hybrid TS+CH for 20 incidents. This method finds improved solutions early but then struggles to converge. For 165 incidents, the moving average from the TS+CH is shown in Figure 7-20b, which shows good results within the first 1000 seconds, but then a failure to converge and increasing objective function values over time. These two scenarios suggest that the parameters of the TS+CH may require further tuning for the dynamic model to be suitable for large scenarios. Although solutions do not converge, a stopping condition of 1000 seconds of CPU time is adequate to find good solutions from the TS+CH heuristic. Figure 7-20c and Figure 7-20d show the moving averages from the ACO+CH.3 solution heuristic for 20 incidents and 165 incidents respectively. It is found that solutions from ACO+CH.3 converge quickly with time to converge affected by the size of the problem.

The TS+CH approach shows improvement in the first iterations but then begins to move away from optimal neighbourhoods, while the ACO+CH.3 is able to converge to a good solution within the time limit. As such, ACO+CH.3 is expected to outperform TS+CH as well as CH for various problem sizes. However, early improvements in incumbent solutions from TS+CH suggest that it may be

appropriate to run this hybrid heuristic before or after the ACO+CH in order to refine solutions.

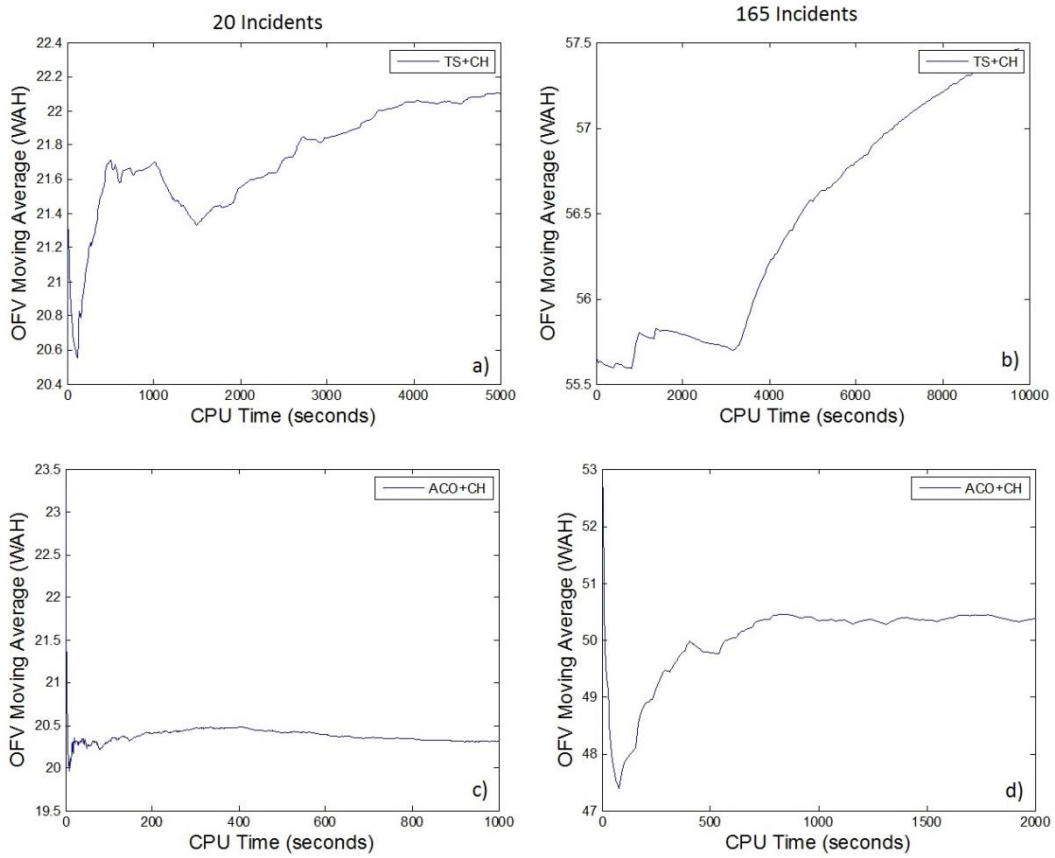


Figure 7-20 Moving average of the objective for scenarios of 20 and 165 Incidents for hybrid TS+CH and hybrid ACO+CH heuristics

7.4.2 Weekly Shift Schedule

The model is solved for one week of incidents for a single horizon covering: the entire week; daily intervals; and hourly intervals, with the CH, TS+CH and ACO+CH.3 heuristics. The stopping condition for each horizon is based on a CPU time of 16,800 seconds, split evenly across each horizon. Each heuristic is expected to take longer than this amount of time for the entire week because of the time required to process data between horizons.

Table 7-6 shows the resulting objective function value, measured in Weighted Ambulance Hours (WAH), represents the costs of running ambulance services, obtained for solving the dynamic model for one week of ambulance incidents. The ACO+CH.3 hybrid heuristic outperforms the CH and hybrid TS+CH heuristics for

all horizon sizes. It can also be seen that a horizon interval of one day produces superior solutions to both horizon intervals of one week and one hour. A weekly horizon contains too much data to be solved effectively in a short amount of time, while an hourly horizon interval does not contain enough information about future ambulance requirements to plan ambulance movements as well as the daily horizon can. The results for five incidents, twenty incidents and the weekly solution also confirm that the dynamic model outperforms the static model presented in Chapter 6.

Table 7-6 Results for solving one week of incidents with the dynamic ambulance scheduling model

	Horizon Length					
	Δt = 10800 mins (1 Week)		Δt = 1440 mins (1 Day)		Δt = 60 mins (1 Hour)	
	solution (WAH)	*cputime (secs)*	*solution (WAH)*	*cputime (secs)*	*solution (WAH)*	*cputime (secs)*
CH	497.87	963.12	494.42	534.09	537.12	2394.94
TS+CH	479.97	18077.07	454.68	17535.93	501.12	18222.73
ACO+CH.3	**454.36**	18219.31	**446.20**	17621.70	**464.53**	20183.10

The best solution (from the ACO+CH.3 heuristic with the daily horizon) is investigated further. In Table 7-7, the performance of the heuristic is investigated. The resulting schedule performs exceptionally well, with an average response time of less than six minutes for emergency incidents and over 95% of emergency incidents receiving a response within 10 minutes. The results for the 50^{th} and 90^{th} percentile of response time for emergency incidents are 4.90 mins and 8.65 mins respectively. This outperforms real percentile response times for emergency incidents. Response times for urgent and non urgent incidents are also good, with 92% of all incidents receiving a response in less than 30 minutes. Dispatch to clear for ambulances in the best schedule for the dynamic model also outperform dispatch to clear times extracted from the incident data. It is unclear how much of this effect is from a reduction time spent travelling and ramping due to decisions informed by the scheduling model, and how much may be an effect of the estimates of travelling time and ramping in the case study. Further scenarios should be tested to confirm this result.

Table 7-7 Performance of the best schedule found with the dynamic model

Priority	**Average response time (mins)**	**Percentile response time (mins)**		**Percentage met in**			**Dispatch to Clear Time (mins)**		
		50th	*90th*	*< 10 mins*	*< 30 mins*	*< 60 mins*	*Average*	*50th Percentile*	*90th Percentile*
ALL	10.73	6.06	23.97	76.72%	92.09%	98.06%	51.00	40.52	91.68
Emergency	5.48	4.90	8.65	95.51%	100.00%	100.00%	59.87	50.85	106.75
Urgent	9.93	6.54	22.27	73.77%	93.63%	100.00%	57.59	49.02	97.67
Non Urgent	15.75	7.31	40.41	63.46%	84.28%	94.89%	38.36	31.87	63.27

Table 7-8 Schedule components for the dynamic model with daily horizons

Solution Heuristic	Ambulances			Ambulance shifts			Average ambulances per shift			Overtime (mins)			Average overtime per shift (mins)		
	Type I	Type II	Type III	Type I	Type II	Type III	Type I	Type II	Type III	Type I	Type II	Type III	Type I	Type II	Type III
CH	53	26	3	160	91	12	7.62	4.33	0.57	2511.14	1747.54	132.86	119.58	83.22	6.33
TS+CH	44	25	3	146	81	12	6.95	3.86	0.57	3194.31	1512.92	96.93	152.11	72.04	4.62
ACO+CH	**44**	**26**	**7**	**133**	**86**	**23**	**6.33**	**4.10**	**1.10**	**2845.31**	**1687.75**	**237.81**	**135.49**	**98.51**	**11.54**

Table 7-9 Schedule components for additional horizon lengths for the ACO+CH

Horizon	Ambulances			Ambulance shifts			Average ambulances per shift			Overtime (mins)			Average overtime per shift (mins)		
	Type I	Type II	Type III	Type I	Type II	Type III	Type I	Type II	Type III	Type I	Type II	Type III	Type I	Type II	Type III
Weekly	40	28	5	136	91	16	6.58	4.33	0.76	2982.73	1952.04	64.85	142.03	92.95	3.09
Hourly	37	34	9	125	106	31	5.95	5.05	1.48	2007.36	2068.79	242.39	95.59	98.51	11.54

The components of solutions are shown in Table 7-8 and Table 7-9. The best solution, found using ACO+CH over daily horizons, returns an average of 11.5 ambulances scheduled for each shift, slightly over half of which are type I ambulances. The greatest amount of overtime is from type I ambulances, due to the greater number of these working. Comparison with other heuristics tested for weekly schedules shows that better solutions were obtained by reducing the number of shifts for the costly type I ambulances and utilising more type III to maintain service levels. The ACO+CH heuristic returned a more balanced schedule in terms of different ambulance types, however, the resulting schedule increased overtime at the same time as decreasing total number of ambulance shifts meaning that each individual ambulance works longer hours to meet demand. Resulting schedules from different horizon lengths for the best performing heuristic (ACO+CH) had several interesting properties. Total overtime increased or decreased for each ambulance type according to the number of ambulance shifts of each type of ambulance. As horizon length was decreased, the number of ambulance shifts for type I ambulances also decreased, and ambulance shifts for type III ambulances increased. Shifts for type II ambulances were fewest for the daily horizon intervals where the best solution was found. It is also observed that overtime per ambulance per shift tends to decrease as the horizon length is decreased, with the exception of type III ambulances in the weekly solution which are not highly utilised.

The shift schedule is shown in Figure 7-21 with the corresponding number of scheduled ambulances available each hour in Figure 7-22. From this figure, it can be seen that daily seasonality is present, in agreement with the demand profile, but that the peak weekend ambulance availability is higher than peak ambulance availability for most weekdays. This suggests that the ACO+CH.3 heuristic may overestimate the number of ambulances required during off-peak times. A ten hour subsection of the ambulance schedule corresponding to Saturday morning ambulance schedules is shown in Figure 7-23. In this schedule, the workload appears reasonably balanced, but utilisation of ambulances is low.

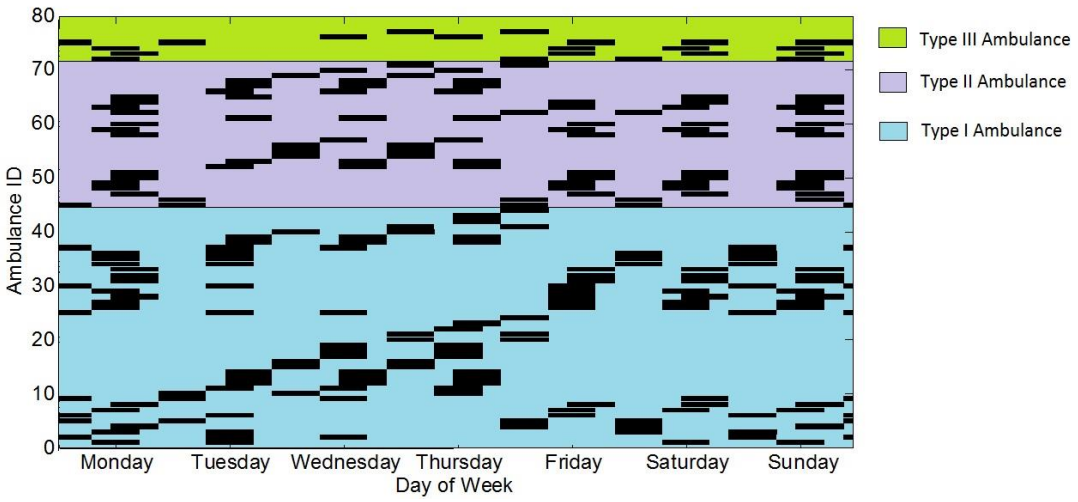


Figure 7-21 Best shift schedule from the dynamic model

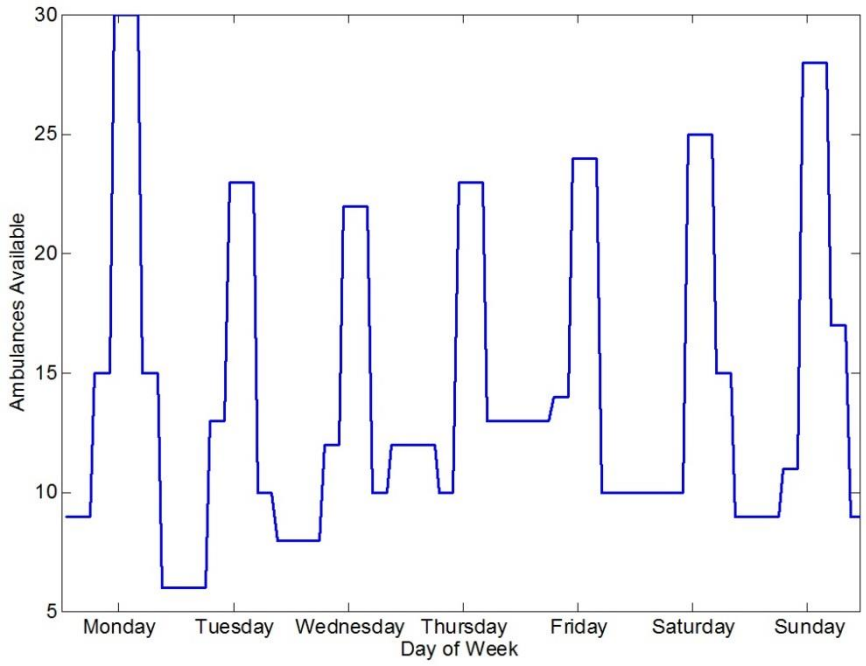


Figure 7-22 Ambulances available each hour from the best schedule in the dynamic model

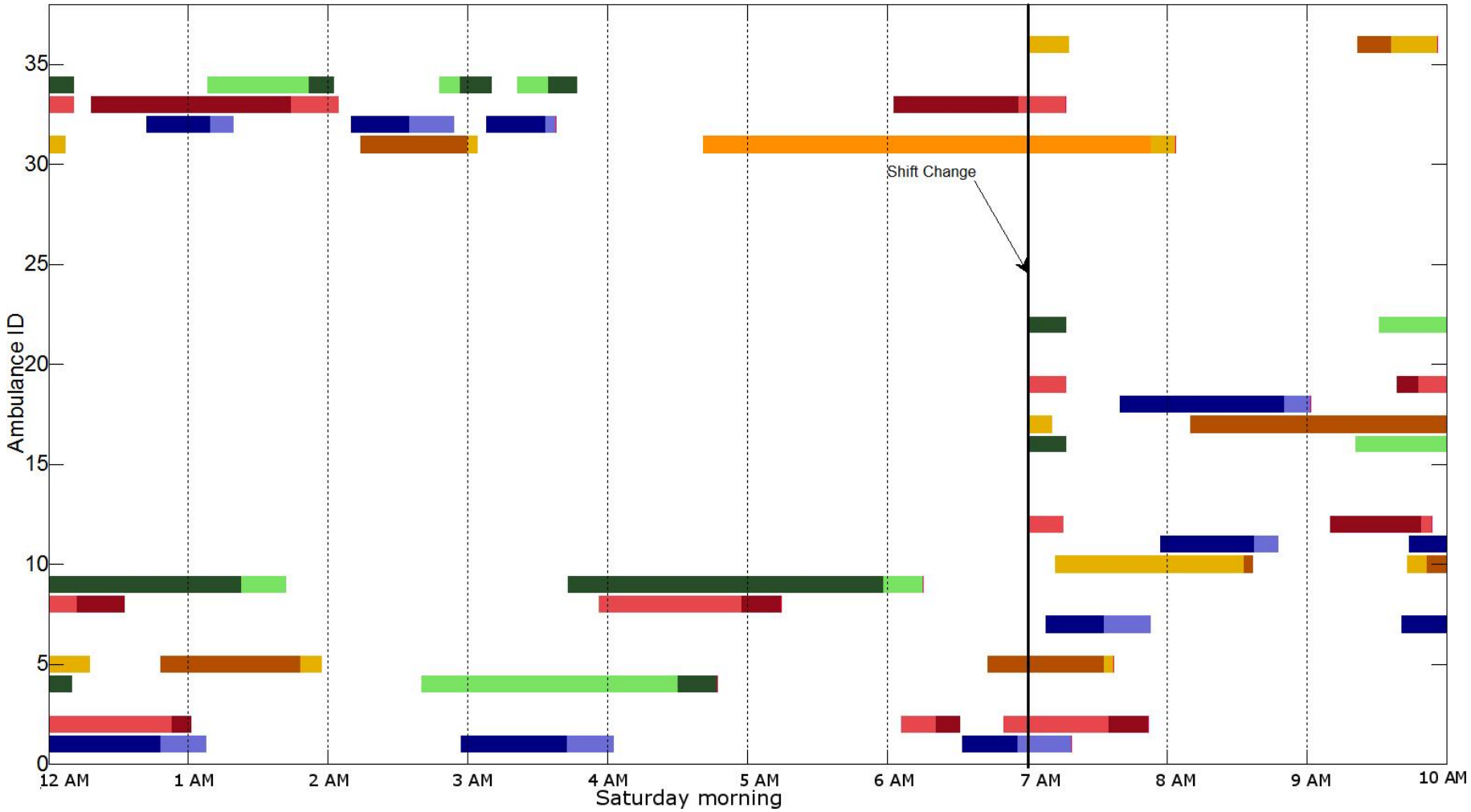


Figure 7-23 Subsection of the schedule covering 10 hours of incidents during off-peak time for the dynamic model

7.4.3 Utilisation of Ambulance Stations

The number of ambulances scheduled to each station is examined in this section. Table 7-10 shows the number of ambulance hours scheduled per station across the week from the best solution. This number of hours utilised is 150% of the number of ambulance hours present in the 2006/07 Workforce data. Increased demand and limited shifts on which ambulances can be scheduled account for part of this gap. However, the scheduling models presented in this thesis require a greater number of ambulances but result in better performance measures, as seen in Section 7.4.2. Future work may modify the values regarding performance measures and due dates in the scheduling models to investigate the impact this would have on the number of ambulances required. It is also evident, from the percentage of ambulances scheduled at each station as shown in Table 7-10, that some ambulance stations receive more ambulances scheduled at their location than others. The information in this table represents the location at which ambulance crews begin and end each shift. This provides support for increasing capacity at certain ambulance stations, such as Chermside. However, future scheduling models may need to include capacity constraints for ambulance stations, as the scheduled number of ambulances may not be able to be physically located at a station.

Table 7-10 Utilisation of ambulance stations in the best solutions from the dynamic model

	1: Northgate	**2: Kedron Park**	**3: Chermside**	**4: Spring Hill**	**5: Roma Street**
Ambulance Hours Scheduled	80	60	1150	190	940
Percentage of ambulances scheduled at station	2.60%	2.60%	49.35%	6.49%	38.96%

7.4.4 Objective Weights Analysis

The objective function uses weights to balance more costly ambulance types and overtime hours compared to regular shift hours. The effects of varying these weights are tested with the ACO+CH.3 heuristic and daily intervals. The same weight variations as were tested in Section 6.3.3 for the static model are tested here. Results given in Table 7-11 indicate very little difference in the components of the solution when weights are varied, indicating robust solutions. Future work suggested includes additional sensitivity analysis to check the robustness of the solutions with respect to objective weights.

Table 7-11 Results from the ACO+CH.3 heuristic for daily horizons with various weights in the objective function

Objective Weights	Solutions (WAH)			Average ambulances per shift			Average overtime per shift for all ambulances (mins)		
	Best	standard	CPUtime (secs)	Type I	Type II	Type III	Type I	Type II	Type III
Additional emphasis on Type I ambulances	640.18	447.91	17377	6.38	4.29	0.71	151.76	75.79	10.05
No overtime weights	419	448.19	17125	7.05	3.52	0.57	155.49	69.68	1.55

7.4.5 Sensitivity to Demand

The model is solved for two additional case studies (Scenario 2, Scenario 3), generated with the same parameters as the case study which returned the results presented above. A fourth case study (Scenario 4) with demand increased by 50% is also tested. These are tested with the most promising heuristics, that is, ACO+CH and TS+CH for daily horizons. The results are shown below in Table 7-12

It is found that all scenarios return emergency repose times between 5 and 6 mins, with better results from the ACO+CH heuristic than the TS+CH heuristic. However, Scenario 4, which has a 50% higher demand for ambulance services, has a drop in response times for all incidents. This means that emergency services are maintained by the model under higher demand but non-emergency services suffer significantly longer response times.

Some differences in the number of ambulance shifts in each resulting schedule is found from different, but similar scenarios. Scenarios 1 and 2 have differences in the number of ambulances scheduled in the ACO+CH solution similar to differences between solutions from the two metaheuristics used to solve Scenario 1. This suggests agreements between the schedules. Scenario 2, however, is able to assign more overtime onto less costly ambulances but uses more overtime. Scenario 3 uses less overtime that either Scenario 1 or 2 but schedules a larger number of Type I ambulances per shift. The scenario itself has 2.5% more ambulances requiring Type I ambulances leading to an average of 2 additional Type I ambulances per shift.

Scenario 4, with 50% higher demand, uses an approximately 8 additional ambulances (4 Type I and 4 Type III) compared to Scenarios 1 and 2. This is 62.4% more ambulances for 50% higher demand. It appears likely that the number of extra

ambulances required increases at a rate slightly higher than the rate of demand.

Table 7-12 Solutions to the dynamic model from additional case studies

Daily Horizons Scenario	Solution Algorithm	Average response time (mins)		Average ambulances per shift			Average overtime per shift from all ambulances (mins)		
		Emergency Incidents	All Incidents	Type I	Type II	Type III	Type I	Type II	Type III
1	ACO+CH	5.48	10.73	6.58	4.33	0.76	142.03	92.95	3.09
	TS+CH	5.97	10.74	8.33	2.90	0.14	152.11	72.04	4.62
2	ACO+CH	5.30	10.45	6.81	3.38	0.24	124.37	156.78	19.30
	TS+CH	5.39	9.32	7.33	3.14	0.24	142.69	162.31	4.20
3	ACO+CH	5.22	10.31	8.48	2.81	0.29	118.68	78.05	5.09
	TS+CH	5.54	10.05	8.52	3.67	0	95.59	72.77	0
4	ACO+CH	5.18	29.93	10.67	3.52	4.76	267.52	138.04	6.43
	TS+CH	5.38	20.80	13.76	3.43	0.09	236.01	91.94	4.19

7.5 VARIATIONS

This section investigates a variation of the dynamic model where the objective function which minimises cost is replaced with an objective function which minimises response times. This requires the relaxation of constraints surrounding performance measures on tardiness and the removal of constraints on ambulance crew schedules in favour of a pre-allocated ambulance crew schedule. The schedule tested is the solution from the initial dynamic model. Investigating this variation examines whether the scheduling can also be effective at minimising response times, which is of interest for the real time model presented in the next chapter.

7.5.1 Parameters

Most parameters from the dynamic model originally formulated remain the same in this variation. The ones which are new or deviate from the previous parameter list are presented here. Parameters which are no longer necessary are identified and the reasons they are now redundant are explained.

Parameters defining subsets of shifts are not required. This is because ambulance crew shifts are now fixed and any parameters introduced previously to aid in the construction of crew schedules are now redundant. The parameters which are removed in the second instance are:

$\boldsymbol{F_w}$ Set of all shifts beginning in week w

$\boldsymbol{G}$ Set of all night shifts

The variable related to decisions placing ambulances on shifts and allocating them to particular ambulance stations is reconsidered as a parameter in this variation of the dynamic model. This ensures that the ambulance crew schedule, which is the main component of the cost of running ambulance services, is fixed.

$$v_{asf} = \begin{cases} 1, & \text{if ambulance } a \text{ scheduled to work shift } f, \text{with home station } s \\ 0, & \text{otherwise.} \end{cases}$$

Weight parameters in the objective function are different in the variation of the dynamic model because the objective itself is changed. Instead of weights related to the cost of scheduling an ambulance crew onto a shift and overtime costs, there are now penalties applied to tardy, and seriously tardy, responses. The new parameters are:

$\acute{\omega}_p$ Penalty applied to incidents of priority p if they are tardy within the first time window.

$\acute{\sigma}_p$ Penalty applied to incidents of priority p if they are seriously tardy beyond the first time window.

With tardy responses now in the objective, it is no longer necessary to constrain the number of incidents that may be tardy. As a consequence, the following parameter has been removed:

$Q_p(t)$ Maximum number of incidents of priority type p that can be tardy at time t

7.5.2 Variables

Two decision variables that define the number of minutes that each incident is tardy are introduced into the model. Two variables are required because different penalties apply if a response is very tardy.

$\tau_i(t)$ = the time (in minutes) after the first response window, up until the second response window, that incident i waits for a response

$\tau_i'(t)$ = the time (in minutes) after the second response window that incident i waits for a response

The variable $q_i(t)$ counting whether incidents are tardy or not has been updated with a new definition in this variation of the dynamic model, and a second variable for tardiness introduced to aid in logical constraints.

$$q_i(t) = \begin{cases} 1, & \text{if incident } i \text{ receives a response between the first and second due da} \\ 0, & \text{otherwise} \end{cases}$$

$$q_i'(t) = \begin{cases} 1, & \text{if incident } i \text{ receives a resposne after the second due date,} \\ 0, & \text{otherwise} \end{cases}$$

The overtime variable $o_{af}(t)$ is now redundant as it no longer forms part of the objective, and can be removed from the model. However, it is desirable for solutions to the model to allow this information to be extracted for the purposes of comparing the results against the solutions from the original version of the dynamic model.

The variable $\psi_{af}(t)$, assigning the first shift of a period of rostered days off, is not required in this dynamic model variation. Ambulance crew schedules are fixed, therefore RDO periods are fixed and this variable is unnecessary.

7.5.3 Objective

The objective for this model is to minimise penalties for tardiness. Each minute that a response to an incident is tardy will have a penalty. Where the response is very tardy, the penalty increases.

Minimise

$$\sum_{p \in \boldsymbol{P}} \acute{\omega}_p \tau_i(t) + \acute{\sigma}_p \tau'_i(t)$$

7.5.4 Constraints

Precedence, continuity and disjunctive constraints remain the same. Each incident can only have a response on one ambulance and shift at a time, must be handled by appropriate ambulances and be sent to appropriate hospitals. Constraints for returning ambulances to stations at the end of a shift remain the same, and the condition that ambulances cannot be dispatched to new incidents after the end of their shift is kept. Location constraints remain the same, as do incident set constraints.

Tardy constraints are relaxed in this variation of the dynamic model. Constraints 7.37, 7.38 and 7.39, restricting arrival times and the number of tardy incidents, are therefore removed. New constraints on tardy responses are introduced to suit the new decision variables:

$$Mq_i(t) \geq r_{ia}(t) - T_i \qquad \forall i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.76}$$

$$\tau_i(t) \geq r_{ia}(t) - T_i + M(1 - q_i(t)) \qquad \forall i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.77}$$

$$Mq'_i(t) \geq r_{ia}(t) - U_i \qquad \forall i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.78}$$

$$\tau'_i(t) \geq r_{ia}(t) - U_i + M(1 - q'_i(t)) \qquad \forall i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{7.79}$$

Constraints for shift scheduling are no longer required because all ambulance crew shift schedules are as parameters rather than as variables requiring constraints. Constraints 7.45, 7.46, 7.47, 7.48, 7.49, 7.50, 7.51 for assigning ambulance crew to shifts according to business rules are all removed.

Symmetry breaking constraint 7.71, dealing with duplicate ambulances, is modified to be suitable for fixed shift schedules. The modified equation is shown in constraint 7.80. This constraint applies when multiple ambulances of the same type are scheduled to the same ambulance station and shift. The first incident to be assigned to any of these duplicate options will be assigned to the ambulance with the lowest index.

$$M \sum_{i \in \boldsymbol{I}} x_{iaf}(t) \geq x_{ia'f}(t) + M\left(v_{asf} + v_{a'sf} - 2\right) \qquad \forall i \in \boldsymbol{I}, f \in \boldsymbol{F}, k \in \boldsymbol{K}, a \in \Lambda_{\boldsymbol{k}}, a' \in \Lambda_{\boldsymbol{k}}: (a' = a + 1) \tag{7.80}$$

The remaining symmetry breaking, non-negativity and integer constraints remain with additional bounds added for the new variables:

$$q_i(t), q_i'(t) \in \{0,1\} \quad \forall\, i \in \boldsymbol{J} \tag{7.81}$$

$$0 \leq \tau_i(t) \leq (U_i - T_i) \quad \forall\, i \in \boldsymbol{J} \tag{7.82}$$

$$0 \leq \tau_i'(t) \leq M \quad \forall\, i \in \boldsymbol{J} \tag{7.83}$$

7.5.5 Solution Approach

The modified dynamic model contains fewer variables than the original dynamic model, due to a fixed number of ambulances and known return-to-station jobs. However, there are still a large number of disjunctive variables, and as such, exact solutions are expected to be nearly as difficult to find for the variation of the dynamic model as for the original formulation. As the original dynamic model performed best with daily horizons solved with the ACO+CH approach, this variation is also solved with daily horizons and ACO+CH. A weekly horizon is also solved to briefly investigate the effect of horizon length. A CH and an ACO approach are both employed in solving the dynamic model as well, for evaluation of the ACO+CH approach. The same time limit as applied for the initial version on the

Table 7-13 Results from the variation of the dynamic model

	Horizon length	Daily			Weekly		
	Solution Approach	CH	ACO	ACO+CH	CH	ACO	ACO+CH
Average response time (mins)	*Emergency incidents*	4.90	11.33	**4.55**	4.87	4.87	4.87
	All incidents	5.18	9.62	5.67	**5.17**	**5.17**	**5.17**
Maximum response time (mins)	*Emergency incidents*	**37.26**	401.58	51.37	**37.26**	**37.26**	**37.26**
	All incidents	**37.26**	401.58	51.37	**37.26**	**37.26**	**37.26**
50th percentile response time (mins)	*Emergency incidents*	4.24	4.53	**4.13**	4.21	4.21	4.21
	All incidents	**4.57**	4.73	4.67	4.58	4.58	4.58
90th percentile response time (mins)	*Emergency incidents*	7.63	9.47	**7.08**	7.56	7.56	7.56
	All incidents	8.50	9.88	9.14	**8.42**	**8.42**	**8.42**
Percent met in < 10 mins	*Emergency incidents*	95.04	90.07	**97.87**	95.27	95.27	95.27
	All incidents	93.88	90.07	92.01	**93.96**	**93.96**	**93.96**
Percent met in < 30 mins	*Emergency incidents*	**99.53**	95.27	**99.53**	**99.53**	**99.53**	**99.53**
	All incidents	**99.85**	96.64	99.03	**99.85**	**99.85**	**99.85**
Percent met in < 60 mins	*Emergency incidents*	**100**	96.69	**100**	**100**	**100**	**100**
	All incidents	**100**	97.31	**100**	**100**	**100**	**100**
Tardy responses (%)	*Emergency incidents*	4.96	9.93	**2.13**	4.73	4.73	4.73
	All incidents	1.57	4.93	**0.90**	1.49	1.49	1.49
Very tardy responses (%)	*Emergency incidents*	**0.47**	4.73	**0.47**	**0.47**	**0.47**	**0.47**
	All incidents	**0.15**	3.06	**0.15**	**0.15**	**0.15**	**0.15**
Overtime (mins)	*Type I*	**2943.25**	7976.07	3931.23	7318.51	7318.51	3423.11
	Type II	**2678.59**	3203.92	3013.56	3551.06	3551.06	2865.70
	Type III	317.48	318.21	243.59	**171.44**	**171.44**	465.01
OBJ (weighted tardy minutes)		570.42	21226.00	524.26	561.23	561.23	561.23
CPU time (seconds)		141.46	16962.44	16961.93	238.00	17137.94	17177.52

dynamic model is applied as the stopping condition for this variation and the same weekly case study as is used to test the model.

The heuristics required minor modifications to be suitable for finding solutions with fixed shift schedules. While it is known that a feasible solution exists for the case study where tardiness performance measures are met with the given shift schedule, greedy heuristics carry a risk of scheduling ambulances to incidents in such a way that later incidents may not be able to be scheduled on any suitable ambulance (due to overlapping with prior assignments) without being delayed beyond acceptable limits, possibly even to a later shift when new ambulances become available. This situation is allowed in the schedule, in place of introducing new ambulances into the schedule, but large tardiness penalties are applied. Varying the order in which incidents are scheduled, by using the ACO and ACO+CH approaches, is expected to converge upon solutions where these large penalties for severely delayed incidents do not occur.

7.5.6 Results and discussion

This section discusses the results from the dynamic model variation, which has the objective of minimising tardiness with fixed resources. The performance measures evaluated include response times for emergency incidents and all incidents, as well as the percentage of tardy responses. Results are shown in Table 7-13. The average response times for emergency incidents can be reduced down to less than five minutes, and over 90% of all incidents can receive an ambulance response in less than ten minutes. This is a significant improvement over current response times, and is achieved by using the resources recommended by the dynamic model efficiently and effectively.

The CH performed better with the single weekly horizon than with daily horizons. The weekly horizon was able to provide equal or better response times on every measure, although this schedule had higher overtime costs. The more complex ACO and ACO+CH heuristics, when applied with the weekly horizon, did not improve the tardiness or response times found with the CH. However, the ACO+CH method was able to meet the same performance with lower overtime costs. While each heuristic found solutions with the same objective function value for the weekly horizon, a better solution was found by using the ACO+CH method for the daily horizon, indicating that the solutions from the weekly horizons are not optimal. As

was the case with the original dynamic model, the size of the problem may prevent efficient searching of the solution space, resulting in poorer solutions.

The daily horizon results using ACO and ACO+CH methods are more interesting. The ACO solution actually resulted in a poorer performance than the simpler CH. The resultant schedule contained a greater percentage of tardy and very tardy responses, and included an exceptionally long response time for at least one emergency incident. Ineffective use of resources with the ACO heuristic meant that appropriate resources were unavailable for incidents when required. However, the ACO+CH method was able to improve response times for emergency incidents at the cost of increased response times for other incidents, and increased overtime. The maximum response time for emergency incidents also increased, suggesting a more skewed distribution of response times. The total number of incidents receiving tardy responses was less than 3%. This measure is important, and is present in the objective function, as ambulance services are time critical.

7.6 IMPLICATIONS AND FURTHER WORK

This model's solution to the case study is a strategic approach to develop a shift schedule using real demand information and allowing relocations to enhance ambulance positioning. This shift schedule can be used to inform the real time model presented in Chapter 7. It is a novel approach, using disjunctive constraints to prevent overlap in ambulance locations as well as overlap in incidents being processed on an ambulance. This allows overtime to be investigated in an ambulance model and allows the use of relocations for the integrated ambulance scheduling and shift scheduling formulation. An ACO+CH heuristic has also been introduced in this chapter and found to be effective for finding solutions to the dynamic ambulance scheduling model.

The model may also be solved with real time data, should such data become available. This would allow a tactical approach with historical data used to predict incidents in order to aid with relocation decisions, and real time information added into the model as it becomes available. The tactical approach indicates whether additional ambulances are required to maintain the performance measure targets.

It is found that the best shift schedule requires more ambulances than are known to be scheduled, but improves the solution found with the static model by

reducing the number of Type I ambulances and the amount of overtime they utilise. The performance of the scheduling model in response time outperforms the response times achieved in the data on which the case study is based. This suggests that the dynamic scheduling model is able to provide a good shift schedule, where the objective is to minimise the resources required to meet performance measures. It is also noted that the ambulance station located at Chermside, near a major public hospital, receives a large number of ambulances scheduled to this location in the solutions. This indicates a use for the model in determining which stations are most important and where it may be worth increasing capacity.

A variation of the dynamic model investigated the consequences of fixing ambulance crew shift schedules and solving a model to minimise tardiness. This variation was able to reduce response times even further, indicating that a scheduling model is effective at ascertaining ambulance schedules where sufficient ambulances are available. Further work could explore this variation of the model in scenarios where ambulance utilisation is increased, either by increasing demand or by reducing the number of ambulances available, to explore the effects of ambulance utilisation on response time.

There is room to extend this model further. While there is some evidence that the solutions are robust with small changes in objective weights, a greater range of weighting values would provide more insight into the robustness of the solution. It is also desirable to increase the size of the case study and test the model with more ambulance stations, more hospitals and for multiple weeks. The hybrid ACO+CH heuristic may be developed further by allowing the parameters, such as the number of ants, to vary with respect to the size of the problem, as the heuristic showed that various parameters performed better on larger or smaller size problems.

Chapter 8: Real Time Model

The final model presented in this thesis is a real time model. The real time model differs from the static and dynamic models previously presented, in that it cannot utilise deterministic data to plan a stable ambulance crew shift schedule in advance. Instead, the ambulance crew schedule is fixed prior to solving the real time model. An extension to the model allowing for changes, such as calling additional ambulance crew for unscheduled shifts in order to meet unexpectedly high demand, would require associated penalty costs and may require the crew schedule to be resolved.

The real time model is the first model presented in this thesis to schedule meal breaks. These are scheduled around other jobs which ambulance crews undertake, with penalty costs applied for missing meal breaks. The FFSS framework is again used to formulate the model because it has been demonstrated that this approach allows multiple types of jobs to be scheduled on a heterogeneous fleet of ambulances where availability and location of ambulances is dependent on time. As in the previous two models, we present the inclusion of overtime in the objective function. Unlike the previous two models, the objective now contains coverage. As shift schedules are fixed, the focus is changed from identifying the minimum number of ambulances required to meet all incidents within performance requirements for response times, to having the most appropriate level of coverage and minimising tardy responses and tardiness. The objective function for the real time model is a multiple criteria objective function, balancing costs and performance for ambulance services.

The remainder of this chapter is set out as follows: Section 8.1 presents the new additions in the model, including the concept of coverage and meal breaks; Section 8.2 presents the formulation of the MIP model; Section 8.3 describes the solution approach, including how data would need to be presented for the real time model; and Section 8.4 concludes the chapter with a summary, implications and further suggested work.

8.1 NEW ADDITIONS IN THE REAL TIME MODEL

Solving the real time model relies on a greater wealth of information being supplied each time it is initialised. This would allow the model to make use of the most accurate information about incident requirements, expected travel times and expected delays at hospitals, and ambulance location and status. The impact of this is that less information is required to be stored in the model variables between time steps, and ambulance shift schedules do not change. In the real time model, certain variables now require fewer dimensions than are present in the static and dynamic models.

This model disallows ambulances to sit idle at hospitals or incident locations for extensive amounts of time. In the real world, an ambulance would not loiter at the scene of an incident past the time it was required to be there. This model reflects that circumstance by directing an ambulance to travel to an ambulance station if it is not required for another job.

Coverage and meal breaks are also accommodated in the real time ambulance scheduling model, and the objective function has been changed as a result.

8.1.1 Coverage

Coverage, for the purposes of the real time model, refers to the area that is sufficiently covered by available resources. An ambulance *a* is considered to cover a node *n* if ambulance a is able to respond to an incident arising in node n within a given time limit. Nodes, depending on the expected level of demand, may require multiple ambulances to provide a sufficient level of coverage. Coverage requirements are dynamic and specify the number of ambulances required at a specific node for a given hour of the week. The requirements are based upon expected demand extrapolated from real data. A coverage gap occurs whenever a node has fewer ambulances covering it than requested in the coverage requirements for that node. Minimising coverage gaps is the same as maximising coverage.

A single coverage radius is selected for the real time model. An available ambulance covers a node if it can be reached within eight minutes (allowing use of lights and sirens) from the last known location of the ambulance. Any ambulance can contribute to coverage, regardless of vehicle type. It is desired to maximise the number of nodes that are sufficiently covered. Double coverage models and coverage

requirements based on ambulance type are left for further work after the real time model is investigated with the single coverage radius for all ambulances.

During times when demand is unexpectedly high, it is unlikely that coverage requirements will be able to be met. Information on coverage is a useful indicator for decision makers that it may be time to call additional ambulances, not currently scheduled to work but able to be called in to work a partial shift. This is a proposed extension to the real time model.

8.1.1.1 Coverage Requirements

Coverage requirements are a dynamic parameter for the real time model. These are defined over a set of spatial nodes within the area in which ambulance services operate. The coverage requirements can be used to solve the real time model when paired with real time information. In this section, coverage requirements are defined, their integration into the model explained and a quick method of estimating coverage requirements is presented.

A grid of nodes for the area of interest may be defined and coverage requirements estimated for each node. As an example, the area of the case study is divided into zones by creating a 7 x 9 grid of 2 km x 2 km squares, and expected demand is used to determine requirements. Coverage requirements are introduced as a new dynamic parameter $\rho_n(t)$, representing an integer number of ambulances desired to be available at node *n* at time *t*. This value represents all ambulance types and is based on all incidents.

Incident arrival rates are extracted for each hour of the week for each defined zone. These are used to determine the number of ambulances required at each location, with respect to changing demand levels throughout the day. A simple process takes the expected number of incident arrivals, assumes an average processing time greater than 60 minutes, so that any ambulance dispatched is not expected to be available again until the next hour, and calculates the number of ambulances required to ensure there will be at least one ambulance available to attend all incidents. The assumption about processing time is supported by results from the dynamic model and by information about ‘dispatch to clear’ times in Section 5.2.5.

The most common requirement for coverage is a single ambulance. At times of peak demand, coverage requirements reach a maximum of four ambulances requested to cover a node (or three if only emergency and urgent demand are considered for determining coverage requirements). No dependency between nodes is considered in the simple model. Dependency should be considered for a more realistic map of coverage requirements. This is because, while a group of nodes may each require a single ambulance, covering this group with a single ambulance may be insufficient to meet total demand arising from this group of nodes. Improvements to the estimation of coverage requirements could be made through a more complex process, possibly involving a queuing model to estimate coverage requirements.

8.1.1.2 Look Ahead Time

The real time model seeks to improve coverage at a time $\hat{t}$ described as the look ahead time. Time $\hat{t}$ is shortly after the time t at which it was initialised. It is necessary to consider coverage at the look ahead time to prepare the system for emergent incidents. Determination of coverage at time $\hat{t}$ requires an investigation of ambulance location and status at time $\hat{t}$, because an ambulance is only available to cover a node if it is available and is able to travel to that node within a limited amount of time. Look ahead constraints are introduced into the real time model and applied to variables indicating ambulance status and location at time $\hat{t}$.

Ambulance status and expected location at the look ahead time must be able to be extracted from information within the model. This can be done by isolating the last event to occur prior to the look ahead time. Three events, for each job, can change the availability of an ambulance. These are dispatch events, occurring at time d_i; ambulance arrival (i.e. response received) events, occurring at time r_i; and clear events, occurring at time c_i. For example, an ambulance will be busy at the look ahead time if the last event prior to time $\hat{t}$ is a response on scene event associated with variable r_i. In this instance, the last known location of the ambulance will be L_i. Similarly, an ambulance will be available at location θ_i at the look ahead time if the last event to occur prior to $\hat{t}$ is a clear event at time c_i.

Figure 8-1 shows an example schedule to illustrate which variable is selected as an indicator of the last event prior to look ahead time $\hat{t}$. In this example, ambulance A1 will be en route to job J2, but is still considered available as it may be

reassigned. Ambulance A2 is busy as it is has begun treatment of a patient connected to job J4. Ambulance A3 is available as it cleared job J6 prior to the look ahead time and has not yet begun another job. Ambulance A4 has no event prior to the look ahead time and will be considered either available or unavailable dependent on the whether a crew is scheduled to be available at time $\hat{t}$. Ambulance availability at the look ahead time must consider the ambulance crew schedule. An ambulance will not be considered available at the look ahead time if it is not due to begin a shift until after time $\hat{t}$. Similarly, when time $\hat{t}$ is after the end of the shift for an ambulance that has returned to its home ambulance station, the ambulance shall no longer be allowed to have an available status.

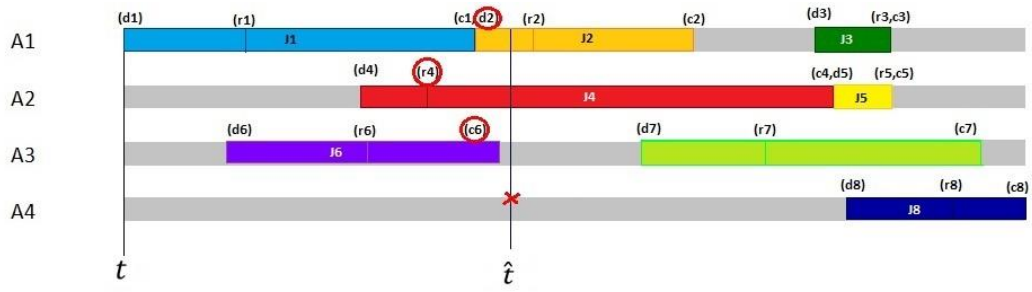


Figure 8-1 Example schedule identifying the last events to occur before the look ahead time for each ambulance

8.1.2 Breaks

As briefly mentioned in Section 5.1.2, two breaks are required to occur every shift. The first is a 30 minute meal break intended to be allocated within a two hour time window between the fourth and sixth hour after an ambulance began a shift. The second is a 20 minute rest break which can occur any time during the shift.

Breaks are included in the real time model as a subset of all jobs to which ambulances must respond. Each break may receive a response from only one ambulance, specified in the input parameters, to ensure that each ambulance is designated the correct number of breaks. Meal and rest break jobs are introduced into the model each time an ambulance begins a new shift and removed from the model's input once they are cleared or the ambulance has ended its shift. Meal breaks may begin at any location and are assumed to end at the same location at which they began each time the real time model is solved. In reality, ambulances are likely to change location during a meal break. The nature of the real time model allows the

current ambulance location to update meal break start and end locations whenever the model is initialised while a break is ongoing.

Constraints applied to meal breaks attempt to place breaks within the best time window and to give the full amount of time allocated for meals. However, responding to incidents and ending a shift take priority, and may pre-empt meal breaks. Skipping or interrupting breaks is an unfavourable outcome within a schedule, as it can lead to fatigued crews on ambulances, but is permitted where an ambulance allocated a break is nearby to an emergent incident that requires an immediate response. Cost penalties currently do not exist for skipping, interrupting or scheduling breaks outside time windows. However, penalties can be applied within the real time model by use of a multiple criteria objective function, with weights applied to indicate the significance of these poor outcomes for allocating breaks.

8.2 FORMULATION

This section highlights the assumptions made in the real time model and outlines the necessary parameters, variables and constraints. The objective value function is also discussed.

8.2.1 Assumptions

The assumptions for the real time model are listed here and explained in further detail beneath this list:

- Availability of ambulances is the availability of ambulance crew (not the ambulance vehicle)
- Different crew mixes have different costs and are able to respond to different sub-sets of all incidents
- Ambulance crew schedules are defined ahead of time
- Each incident is a job which requires five operations to be completed after dispatch
- Precedence relationships between operations must be obeyed
- Overtime is not limited
- Overtime costs double the per-minute cost of regular time

- Overtime is paid by the minute, not in blocks
- Ambulances must return to their home ambulance station to end a shift
- Overtime is accrued if, and only if, ambulances are not available at their home ambulance station at the designated end of their shift
- Incidents receive exactly one ambulance
- All ambulances are capable of transferring patients to hospital
- Hospital preferences must be met under all circumstances
- Ramping time at a hospital is independent of the number of patients arriving at the hospital by ambulance.
- Pre-emption is permitted, but only during certain operations
- Meal and rest breaks should be scheduled but can be interrupted without being resumed
- Ambulances may be assigned to wait at any ambulance station in response to changes in coverage
- A response is tardy if an ambulance does not arrive by the first due date specified in Section 5.3.2.4
- Ambulances may respond to new incidents even after the time they were due to end a shift

Constraints enforcing that performance measures meet targets are relaxed in the real time model. Ambulances may be dispatched past the time they were due to end the shift if they are still present in the system. Response times for incidents now have neither an upper limit on tardiness nor a limit on the number of incidents that may be tardy. Tardiness is now present in the objective function instead. Overtime may continue to be included in a multiple criteria objective function with appropriate weights. Coverage variables are introduced into the objective function with new constraints affecting the values that these may take.

Unlike the static and dynamic models, the real time model is not strategic and does not build a shift schedule. Instead, a segment of the shift schedule as determined from the dynamic model is selected and may be used to place scheduled

starting and ending times for each ambulance directly into the model as parameters. Ambulance shifts and station allocation are now input, as is the location of each ambulance at the time when the real time model is called. The real time model is formulated such that it can solve the ambulance schedule for the near future, beginning at time *t* when it is initialised. The period of time considered is small enough that it is safe to assume that each ambulance in the model will only have one start and finish time during the period considered. The initial version of the real time model assumes a fixed shift schedule and does not allow for additional ambulances to be called.

The benefit of the real time approach is to keep the number of variables low so that the problem size is small enough to find good solutions quickly. Jobs which are not anticipated at the time the model is called, or are pre-planned for later shifts, will not form part of the ambulance schedule. The number of interval variables for the jobs that are present in the real time model is reduced from the equivalent number of variables in the dynamic model. This is through elimination of the ambulance dimension for dispatch, arrival and clear time variables, which is possible because shift schedules are fixed. Ambulance and incident status variables used in continuity constraints in the dynamic model are also not required in the real time model, as status becomes a parameter queried during the initialisation of the model. This allows the real time model to be simplified to remove dependence on *t*. The input and output for the real time model is illustrated in Figure 8-2.

8.2.2 Parameters

The real time model is initialised with the time *t* when the model is called. Any new decisions resulting from the real time model cannot begin any earlier than time *t*. A second time parameter $\hat{t}$ is also defined as the 'look ahead time'. Coverage in the model is determined for the look ahead time in order to make use of relocations that will better prepare the system for incidents arising in the near future. Time dependent coverage requirements are calculated in advance for each node in a grid that covers the entire area considered in a problem. The appropriate coverage requirements are called as input for the real time model.

t	The time at which the real time model is called
$\hat{t}$	The look ahead time used for optimising coverage

ρ_n The number of ambulances required to cover each node *n*

Logical constraints in the model also require a parameter with a value much larger than that which can be obtained by any of the decision variables. Previously, this has been defined as greater than the ending time of the final shift in the model. As the set of shifts is not directly present in the real time model, this value must be chosen more cautiously. The value chosen for logical constraints in the real time model is time *t* plus 1 week (in minutes).

M Large value for logical constraints: $M = t + 10080$

The parameters in the model, as with the static and dynamic models, must include information about hospitals and ambulance stations.

Hmax Total number of hospitals

H Set of all hospitals {1..Hmax}

Smax Total number of ambulance stations

S Set of all ambulance stations {1..Smax}

Priority types for incidents and vehicle types for ambulances remain the same in the real time model as in previous models developed in this thesis.

Pmax Total number of incident priority types (i.e. triage categories)

P Set of all priority types {1..Pmax}

Kmax Total number of different ambulance vehicle types

K Set of different ambulance vehicle types {1..Kmax}

Only ambulances present at the beginning of time *t* plus those that become available in the near future are included in the real time model input. As with previous models, there is a sub-set for ambulances of each vehicle type. Unlike the previous models, the present location of each ambulance is queried whenever the real time model is initialised.

Amax Total number of ambulances available

A Set of all available ambulances {1..Amax}

$\boldsymbol{\Lambda_k}$ Set of all available ambulances of vehicle type *k*

$\boldsymbol{\Gamma_a}$ Location of ambulance *a* when the model is initialised at time *t*

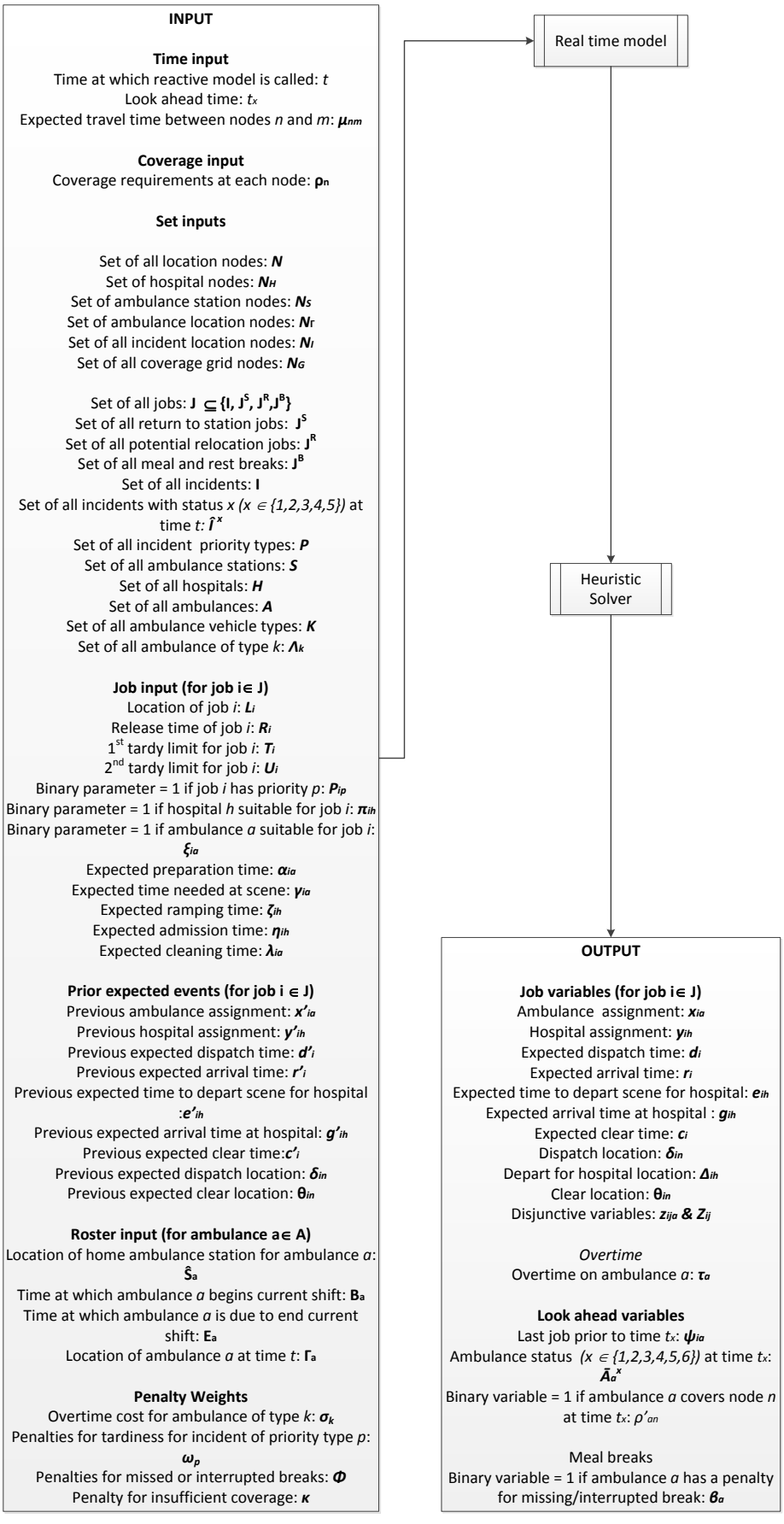


Figure 8-2 Input parameter and output values for the real time model

Ambulance crew shift schedules in the real time model are defined by a scheduled beginning time and a scheduled ending time for the availability of each ambulance *a*.

B_a Time at which ambulance *a* becomes available on the current shift

E_a Time at which ambulance *a* is due to end the current shift

Incidents, meal breaks, potential relocation and return-to-station jobs must each be defined as separate sub-sets within the set of all jobs present in the model. These jobs have some overlapping constraints and so it is desirable to define them all within a single set. Sub-sets become necessary as there are some constraints which only apply to a particular type of job.

Jmax Total number of known jobs not yet clear

$\boldsymbol{J}$ Set of all known jobs not yet clear {1..Jmax}

Imax Set of all known incidents not yet clear

$\boldsymbol{I}$ Set of all known incidents not yet clear (1..Imax}

$\hat{\boldsymbol{I}}^1$ Set of all incidents still awaiting an ambulance to be dispatched at time t when the real time model is initialised

$\hat{\boldsymbol{I}}^2$ Set of all incidents still waiting for a dispatched ambulance to arrive at the scene of the incident when the real time model is initialised

$\hat{\boldsymbol{I}}^3$ Set of all incidents where an ambulance is still present at the scene of the incident when the real time model is initialised

$\hat{\boldsymbol{I}}^4$ Set of all incidents where an ambulance has left the scene of the incident and is en route to a hospital when the real time model is initialised

$\hat{\boldsymbol{I}}^5$ Set of all incidents still being processed at a hospital when the real time model is initialised

$\boldsymbol{J^S}$ Set of all jobs required to return ambulances to their home stations

$\boldsymbol{J^R}$ Set of all potential relocation jobs

$\boldsymbol{J^{B1}}$ Set of all meal break jobs with duration of 30 mins that are still to be taken, with preference to schedule between 4 and 6 hours after beginning a shift

$\boldsymbol{J^{B2}}$ Set of all meal break jobs with duration of 20 mins that are still be taken sometime during the shift

Each job has associated parameters, for example the location at which an incident requires a response is the destination parameter in this model. Parameters

are called each time the real time model is initialised and, in the event that incident requirements or expected processing times change, they are updated the next time the real time model is called. Incidents have a larger number of parameters than meal break, return-to-station and relocation jobs. Only ambulance suitability and location parameters are required for non-incident jobs. Meal breaks and return-to-station jobs will have exactly one ambulance which is suitable to attend to job (i.e. $\sum_{a \in \boldsymbol{A}} \xi_{ia} = 1 \ \forall i \in \{\boldsymbol{J^S}, \boldsymbol{J^{B1}}, \boldsymbol{J^{B2}}\}$). Relocation jobs can be assigned to any ambulance so that $\xi_{ia} = 1 \ \forall i \in \boldsymbol{J^R}, a \in \boldsymbol{A}$. The destination for relocation jobs is the location of the ambulance station that is the destination for each of these jobs. The destination for return-to-station is the home ambulance station from the shift schedule. Meal breaks are not required to have a location in this model, although it is feasible to enforce the home ambulance station as a location for breaks in future versions of this model.

ξ_{ia} $= \begin{cases} 1, \text{if ambulance } a \text{ is suitable to respond to job } i, \\ 0, \text{otherwise} \end{cases}$

L_i Destination of job i (other than meal break jobs)

R_i Release time of incident i

T_i Tardy response time for incident i

U_i Upper bound on arrival time for incident i

P_{ip} $= \begin{cases} 1, \text{if incident } i \text{ has priority type } p, \\ 0, \text{otherwise} \end{cases}$

π_{ih} $= \begin{cases} 1, \text{if hospital } h \text{ is suitable to receive job } i, \\ 0, \text{otherwise} \end{cases}$

α_{ia} Expected time for ambulance a to prepare for a response to incident i

γ_{ia} Expected time for ambulance a to handle incident i at the scene or expected remaining time off duty for break job i on ambulance a

ζ_{ih} Expected time that incident i will spend ramping at hospital h

η_{ih} Expected time for incident i to be passed onto/admitted into hospital h

λ_{ia} Expected time for cleaning ambulance a after responding to incident i

It is possible that jobs may have been assigned to ambulances and have begun processing prior to the real time model being solved. To ensure this information is used in the real time model, it must also be called for each relevant job.

$\check{d}_i$ Expected dispatch time of job i prior to time t

$\check{r}_i$ Expected arrival time of incident i prior to time t

$\check{c}_i$ Expected clear time of job i prior to time t

$\check{e}_{ih}$ Expected time that incident *i* begins transportation to hospital *h* prior to time *t*

$\check{g}_{ih}$ Expected time that incident *i* completes transportation to hospital *h* prior to time *t*

$\check{x}_{ia}$ $= \begin{cases} 1, \text{if incident } i \text{ is assigned to ambulance } a \text{ prior to time } t \\ 0, \text{otherwise} \end{cases}$

$\check{y}_{ih}$ $= \begin{cases} 1, \text{if incident } i \text{ is assigned to hospital } h \text{ prior to time } t \\ 0, \text{otherwise} \end{cases}$

A road network is not defined in the real time model. Instead, nodes present in the model include all locations which ambulances may visit (that is, starting and ending locations for any possible ambulance assignment), existing locations of ambulances and an additional grid of nodes to be used when determining coverage. Estimated travel times between each node are queried when the model is initialised. Location parameters for ambulance stations and hospitals are static; however, locations for incidents depend on which incidents are present in each horizon. Ambulance locations at the time the model is called ($\boldsymbol{\Gamma_a}$) are also nodes.

$\boldsymbol{N_G}$ Set of all coverage nodes forming the grid over which coverage is determined

$\boldsymbol{N_H}$ Set of all location nodes at hospitals

$\boldsymbol{N_S}$ Set of all location nodes at stations

$\boldsymbol{N_I}$ Set of all location nodes at incident scenes

$\boldsymbol{N_A}$ Set of all location nodes for ambulances not at a hospital, station or incident scene at time *t*

$\boldsymbol{N}$ Set of all location nodes ($\boldsymbol{N \subseteq \{ N_I, N_S, N_H, N_A \}}$)

$\mu_{l_1 l_2}$ Expected travel time from location l_1 to l_2 at time *t*

Weights applied in the objective function are commonly defined by ambulance type or priority type. This allows flexibility in the model.

σ_k Weights applied to each ambulance type *k* to represent cost of overtime in the objective function

ω_p Weights applied to tardiness for each incident of priority type *p*

ω'_p Weights applied to tardiness beyond the upper limit for each incident of priority type *p*

Φ^1 Penalty weight applied to untaken meal breaks

Φ^2 Penalty weight applied to untaken rest breaks

$\widehat{\Phi}^1$ Penalty weight applied to interrupted meal breaks or meal breaks taken outside the designated time window

$\widehat{\Phi}^2$ Penalty weight applied to interrupted rest breaks

κ Penalty weight applied for insufficient coverage

8.2.3 Variables

Decision variables for this model are the variables appearing in the objective function, whilst other variables, subject to constraints, affect the values available to each decision variable.

8.2.3.1 Decision Variables

There are multiple objectives for the real time model: minimising tardiness, maximising coverage, minimising overtime and minimising penalties for missing or interrupting breaks. The objective can be decomposed into each of the components by varying the weights defined in the parameters section. The components themselves are defined by the following variables:

- A coverage gap variable minimises the number of location nodes with insufficient ambulances nearby.

  ρ'_n percentage of the coverage requirements for node *n* that remains unmet.

- Two tardiness variables determine the number of minutes by which an incident is either tardy or very tardy. A maximum value is placed on the first tardiness variable to prevent a minute of tardy time being counted twice, once for regular tardiness and once for exceptional tardiness. The second tardiness variable will only be positive if the first tardiness variable is positive and has reached the maximum allowed value.

  τ_i number of minutes that incident *i* is considered tardy.

  τ'_i number of minutes that incident *i* is considered very tardy.

- Overtime may be included in the objective through an overtime decision variable that applies to each ambulance.

  o_a minutes of overtime accrued on ambulance a.

- The real time model is the first model presented in this thesis to include meal break and rest break jobs. Penalties are applied in the objective function if breaks are not taken, are interrupted before they are due to clear or are not taken within the prescribed time window. These penalties are applied through the use of decision variables determining whether breaks were taken or not and, if they were taken, whether they were completed or interrupted.

$$\beta_a^1 = \begin{cases} 1, \text{if meal break , on ambulance } a, \text{is interrupted} \\ \quad \text{or is taken outside the designated time window} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{\beta}_a^1 = \begin{cases} 1, \text{if meal break, on ambulance } a, \text{is not taken} \\ 0, \text{otherwise} \end{cases}$$

$$\beta_a^2 = \begin{cases} 1, \text{if rest break, on ambulance } a, \text{is interrupted} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{\beta}_a^2 = \begin{cases} 1, \text{if rest break, on ambulance } a, \text{is not taken} \\ 0, \text{otherwise} \end{cases}$$

8.2.3.2 Dependent Variables

Decision variables are minimised in the objective function while dependent variables, necessary for modelling real world assumptions, interact with the decision variables through a series of constraints.

Variables for the real time model assign jobs to ambulances and incidents (a sub-set of all jobs) to hospitals. Unlike the dynamic and static models, it is not necessary to assign incidents to shifts. Each ambulance to which it may be assigned already has a nominated shift and home ambulance station.

$$x_{ia} = \begin{cases} 1, \text{if incident } i \text{ is assigned to ambulance } a \\ 0, \text{otherwise} \end{cases}$$

$$y_{ih} = \begin{cases} 1, \text{if incident } i \text{ assigned to hospital } h \\ 0, \text{otherwise} \end{cases}$$

Disjunctive variables determine the order in which jobs receive responses if they are on the same ambulance. Two disjunctive variables are used. One variable contains information on jobs which are adjacent on the same ambulance. The other dependent variable contains information on the order of all jobs assigned to specific ambulance a.

$$Z_{ij} = \begin{cases} 1, \text{if incident } i \text{ immediately preceded incident } j \\ \quad \text{on the same ambulance} \\ 0, \text{otherwise} \end{cases}$$

$$z_{ija} = \begin{cases} 1, \text{if incident } i \text{ precedes incident } j \text{ on ambulance } a \\ 0, \text{otherwise} \end{cases}$$

Variables for the beginning and ending times of each job are necessary within the FFSS formulation. For incidents, the time when the ambulance arrives at the scene and the times when ambulances begin and end transportation to a hospital are also required. In the real time model, it is not required to specify the ambulance on which each of these events occurs.

d_i dispatch time of job *i*

r_i arrival time of job *i*

c_i clear time of job *i*

e_i time at which incident *i* begins transportation to hospital

g_i time at which incident *i* completes transportation to hospital

Dispatch and clear locations for each job are also a key component of the integrated scheduling model. These allow disjunctive constraints to prevent ambulances being in two places at the same time as well as preventing ambulances from processing two jobs at the same time. The real time model introduces a new location variable for incidents, defining the node at which an ambulance is located when travel to hospital begins.

$$\delta_{in} = \begin{cases} 1, \text{if job } i \text{ is dispatched from node } n \\ 0, \text{otherwise} \end{cases}$$

$$\Delta_{in} = \begin{cases} 1, \text{if incident } i \text{ begins travel to hospital from node } n \\ 0, \text{otherwise} \end{cases}$$

$$\theta_{in} = \begin{cases} 1, \text{if job } i \text{ is due to be cleared at node } n \\ 0, \text{otherwise} \end{cases}$$

The real time model seeks to minimise coverage gaps in the objective function. This requires the coverage at the look ahead time to be monitored in the model. Ambulance status and location at the look ahead time determine which nodes each ambulance a covers. A set of dependent variables is created to identify the ambulance status and last event on each ambulance a at the look ahead time.

$$\widehat{\Psi}_{ia} = \begin{cases} 1, \text{if job } i \text{ is the last job to begin prior to } \hat{t} \text{ on ambulance } a \\ 0, \text{otherwise} \end{cases}$$

$$\hat{\chi}_{ia} = \begin{cases} 1, \text{if job } i \text{ begins prior to look ahead time } \hat{t} \text{ on ambulance } a \\ 0, \text{otherwise} \end{cases}$$

$$\hat{A}_a^1 = \begin{cases} 1, \text{if ambulance } a \text{ has a dispatch event as last event before } \hat{t} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{A}_a^2 = \begin{cases} 1, \text{if ambulance } a \text{ has a response arrived event as last event before } \hat{t} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{A}_a^3 = \begin{cases} 1, \text{if ambulance } a \text{ has a clear event as last event before } \hat{t} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{A}_a^4 = \begin{cases} 1, \text{if ambulance } a \text{ had no event recorded as last} \\ \quad \text{event before } \hat{t} \text{ and is available at time } \hat{t} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{A}_a^5 = \begin{cases} 1, \text{if ambulance } a \text{ has no event recorded as last event} \\ \quad \text{before } \hat{t} \text{ and is not available until after } \hat{t} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{A}_a^6 = \begin{cases} 1, \text{if ambulance } a \text{ has completed its shift and} \\ \quad \text{become unavailable prior to time } \hat{t} \\ 0, \text{otherwise} \end{cases}$$

$$\hat{\rho}_{an} = \begin{cases} 1, \text{if ambulance } a \text{ covers node } n \text{ at look ahead time } \hat{t} \\ 0, \text{otherwise} \end{cases}$$

8.2.4 Objective

The objective function balances performance penalties for the ambulance services. Schedules incur penalties for interrupted or untaken meal and rest breaks, working overtime, tardy responses to incidents and not meeting coverage requirements. These are represented in four weighted terms in a multiple criteria objective function. The model allows any of these weights to be set to zero to focus on selected areas where performance can be optimised.

1st term: Tardiness

The previous models presented in this thesis build a shift schedule for ambulance crews that ensure performance requirements for tardiness and tardy response are met for a deterministic data set. The real time model minimises total tardiness. Tardiness is considered in two phases using the two tardy limits (T_i and U_i) for each incident that were used to constrain arrival times in the previous models. Penalties for arrival times greater than the upper limit (U_i) are greater than penalties for an arrival that is only slightly tardy. This is to discourage extreme tardiness occurring when tardiness is unable to be avoided. Tardiness is also weighted with incident priority type so that schedules will prioritise less tardiness on emergency and urgent incidents than non-urgent incidents. Tardiness is represented in the first term of the objective function.

2nd term: Coverage

Coverage gaps, defined as a lack of an appropriate level of coverage at node *n*, are minimised in the second term of the objective function. A fully covered area will have a coverage gap equal to zero across all nodes. Minimising coverage gaps instead of maximising coverage allows all terms in the objective function to remain

positive and will not maximise coverage beyond what is required. The model assumes that all nodes have equal weights for coverage and may be fully, partially or not at all covered.

3rd term: Overtime penalties

Overtime penalties are counted in the objective function for the real time model as per the objective functions in the static and dynamic models. The number of units (measured in minutes) of overtime for ambulance *a* contribute to the objective function value according to the vehicle type of ambulance *a*. The difference with the real time model is the weighting value. In the previous models, the objective function related to costs through Weighted Ambulance Hours and balanced overtime costs against the cost of placing ambulances onto an entire shift. The shift schedule is now fixed as input in the model and the objective seeks to optimise performance. Overtime weights are adjusted for balance against tardiness and coverage gaps in the new objective.

4th term: Break penalties

Two types of breaks for each ambulance shift are considered in the real time model. Each of these can incur penalties for being interrupted or missed. Additionally, meal breaks can also incur penalties if the break is not interrupted but is scheduled outside the preferred time window. The same decision variable applies penalties for interrupted meal breaks and meal breaks outside the preferred time window, so that a poorly scheduled meal break will only incur the penalty once if both circumstances are true. Weights for the two types of breaks are allowed to be different in the objective function, in order to allow a higher priority to be placed onto meal breaks than rest breaks if desired. Missed breaks incur a higher penalty than inadequate breaks, to model the realistic assumption that a short break at the wrong time is still better than no break.

The objective function for the real time model is a minimised weighted sum of the four terms discussed above. Any of the weights may be set to zero to decompose the criteria in the objective.

Minimise $\sum_{p\in \boldsymbol{P}} \sum_{i\in \boldsymbol{I}} P_{ip}\left(\omega_p\, \tau_i + \omega'_p \tau'_i\right) + \kappa \sum_{n\in \boldsymbol{N_G}} \rho'_n + \sum_{k\in \boldsymbol{K}} \sum_{a\in \Lambda_k} \sigma_k o_a +$

$\sum_{a\in \boldsymbol{A}}\left(\beta_a^1\, \Phi^1 + \widehat{\Phi}^1\, \hat{\beta}_a^1 + \beta_a^2\, \Phi^2 + \widehat{\Phi}^2\, \hat{\beta}_a^2\right)$

8.2.5 Constraints

Precedence constraints

The dispatch time for each job is affected by the following precedence constraints each time the real time model is solved.

Constraint (8.1): The dispatch time for a job cannot be prior to it being released, unless no ambulance was assigned to the job:

$$d_i \geq R_i - M(1 - x_{ia}) \qquad \forall i \in \boldsymbol{J} \tag{8.1}$$

Constraint (8.2): The dispatch time for a job must not occur before the assigned ambulance is available:

$$d_i \geq B_a - M(1 - x_{ia}) \quad \forall i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.2}$$

All types of jobs require a constraint relating dispatch time to time *t* when the real time model is initialised. This constraint takes a slightly different form depending on the type of job.

Constraint (8.3): The dispatch time for an incident *i* which is new to the system (that is, it has not previously been assigned an ambulance or dispatch time) must be greater than or equal to the time *t* at which information is received about the job. For incidents, this constraint is limited to incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{1}}$. The dispatch time for return-to-station jobs is also updated to time *t:*

$$d_i \geq t \quad \forall i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{1}}, \boldsymbol{J}^{\boldsymbol{S}}\} \tag{8.3}$$

Constraint (8.4): Relocation and break jobs must be introduced (or reintroduced if continuing with appropriately updated processing times) with a dispatch time greater than or equal to *t*. This is similar to constraint 8.3, however, these jobs will not always be assigned every time the model is solved and an extra term is required in the constraint:

$$d_i \geq t - M(1 - x_{ia}) \quad \forall i \in \{\boldsymbol{J}^{\boldsymbol{R}}, \boldsymbol{J}^{\boldsymbol{B1}}, \boldsymbol{J}^{\boldsymbol{B2}}\} \tag{8.4}$$

Constraint (8.5): The dispatch time for incident *i*, where a decision is made to reassign ambulances from the previous solution, is required to be no earlier than time *t* when the reassignment decision is made. This constraint only applies to incidents where it is still allowable to reassign ambulances (that is, incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{2}}$):

$$d_i + M(1 - x_{ia}) \geq t - M\,(1 - \check{x}_{ia'}) \quad \forall i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{2}}\}, a \in \boldsymbol{A}, a' \in \boldsymbol{A} \setminus \{a\} \tag{8.5}$$

Constraint (8.6): The dispatch time for any incident *i* where an ambulance arrived on scene prior to time *t* must retain the same dispatch time in the new solution. This applies to incidents with a status of $\hat{\boldsymbol{I}}^{\boldsymbol{3}}$, $\hat{\boldsymbol{I}}^{\boldsymbol{4}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{5}}$:

$$d_i \geq \check{d}_i \quad \forall i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{3}}, \hat{\boldsymbol{I}}^{\boldsymbol{4}}, \hat{\boldsymbol{I}}^{\boldsymbol{5}}\} \tag{8.6}$$

The arrival time of the ambulance for each incident is dependent on dispatch time and travel time. This variable may be updated as a reaction to new information at any time until the ambulance reaches the destination. After reaching the scene of the incident, the arrival time must remain the same.

Constraint (8.7): The expected arrival time of each incident *i* must be greater than or equal to the time of dispatch plus the time for preparation and travel required for the ambulance that has been dispatched. This constraint only applies to incidents still awaiting a response (that is, incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{1}}$ and $\hat{\boldsymbol{I}}^{\boldsymbol{2}}$):

$$r_i \geq d_i - M(1 - x_{ia}) + \alpha_{ia} + \sum_{n \in \boldsymbol{N}} \delta_{in} \mu_{nl_i} \quad \forall\, i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{1}}, \hat{\boldsymbol{I}}^{\boldsymbol{2}}\}, a \in \boldsymbol{A} \tag{8.7}$$

Constraint (8.8): Once a response has arrived at the scene of the incident (that is, an incident *i* has status $\hat{\boldsymbol{I}}^{\boldsymbol{3}}$, $\hat{\boldsymbol{I}}^{\boldsymbol{4}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{5}}$) then the arrival time must retain the same value:

$$r_i \geq \check{r}_i \quad \forall i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{3}}, \hat{\boldsymbol{I}}^{\boldsymbol{4}}, \hat{\boldsymbol{I}}^{\boldsymbol{5}}\} \tag{8.8}$$

Incidents may require transfer to hospital. Transfer of an incident to a hospital cannot begin prior to that incident being ready for transfer. That is, transfer to hospital for incident *i* must be after the ambulance arrival at the scene of incident *i*, plus the required processing time for treatment at the scene. In the event of reassignment to a different hospital, transfer time must be greater than the time when the reassignment decision was made.

Constraint (8.9): Where an incident is yet to complete treatment at the scene of the incident (that is, incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{1}}$, $\hat{\boldsymbol{I}}^{\boldsymbol{2}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{3}}$) then the time when transfer to hospital begins is a simple precedence relation from arrival time and expected treatment time at the scene:

$$e_i \geq r_i + x_{ia}\, \gamma_{ia} \quad \forall i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{1}}, \hat{\boldsymbol{I}}^{\boldsymbol{2}}, \hat{\boldsymbol{I}}^{\boldsymbol{3}}\}, a \in \boldsymbol{A} \tag{8.9}$$

Constraint (8.10): Where incident *i* has commenced, but not completed, travel to a hospital (that is, incidents with status $\hat{\boldsymbol{I}}^{\mathbf{4}}$) then it will retain the same transfer time as in previous solutions if and only if it continues travel to the same hospital:

$$e_i + M(1 - y_{ih}) \geq \check{e}_{ih} \quad \forall i \in \hat{\boldsymbol{I}}^{\mathbf{4}}, h \in \boldsymbol{H} \tag{8.10}$$

Constraint (8.11): It is possible to redirect incident *i* to a different hospital than was selected in the previous solution while it is still en route to the hospital. However, the new time at which transfer to hospital begins cannot be earlier than time *t*. This constraint applies only to incidents currently en route to a hospital (that is, incidents with status $\hat{\boldsymbol{I}}^{\mathbf{4}}$) where the hospital assignment in the current solution is different to the hospital assignment from the previous solution:

$$e_i + M(1 - y_{ih}) \geq t - M(1 - \check{y}_{ih'}) \quad \forall i \in \hat{\boldsymbol{I}}^{\mathbf{4}}, h \in \boldsymbol{H}, h' \in \boldsymbol{H} / \{h\} \tag{8.11}$$

Constraint (8.12): Once an incident has arrived at a hospital (that is, incident status is $\hat{\boldsymbol{I}}^{\mathbf{5}}$), the time at which an ambulance is recorded as beginning transfer to hospital will remain the same as previously recorded:

$$e_i \geq \check{e}_{ih} \quad \forall i \in \hat{\boldsymbol{I}}^{\mathbf{5}}, h \in \boldsymbol{H} \tag{8.12}$$

Constraint (8.13): The arrival time at the hospital is dependent on the time at which an ambulance began transfer plus travel time from the location where transfer began. These times are updated in each solution for incidents with status $\hat{\boldsymbol{I}}^{\mathbf{1}}$, $\hat{\boldsymbol{I}}^{\mathbf{2}}$, $\hat{\boldsymbol{I}}^{\mathbf{3}}$ or $\hat{\boldsymbol{I}}^{\mathbf{4}}$ where the incident has not yet arrived at a hospital:

$$g_i + M(1 - y_{ih}) \geq e_i + \Delta_{in}\mu_{nl_h} \quad \forall i \in \{\hat{\boldsymbol{I}}^{\mathbf{1}}, \hat{\boldsymbol{I}}^{\mathbf{2}}, \hat{\boldsymbol{I}}^{\mathbf{3}}, \hat{\boldsymbol{I}}^{\mathbf{4}}\}, h \in \boldsymbol{H}, n \in \boldsymbol{N} \tag{8.13}$$

Constraint (8.14): Once an incident has arrived at the hospital, the arrival time is fixed in further solutions:

$$g_i \geq \check{g}_{ih} \quad \forall i \in \hat{\boldsymbol{I}}^{\mathbf{5}}, h \in \boldsymbol{H} \tag{8.14}$$

Clear times for incident *i* must always be greater than or equal to the arrival time plus time spent on scene. Additionally, the clear time for incidents requiring transfer to hospital is no earlier than the arrival time at hospital *h* plus the time spent at the hospital. Clear times may change each time the real time model is solved up until the point at which incident *i* is completely cleared.

Constraint (8.15): The clear time for incident *i* cannot be prior to arrival time plus the treatment time at the scene and cleaning time expected for ambulance *a*. While this applies for all incidents, it is only necessary to ensure it is applied to

incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{1}}$, $\hat{\boldsymbol{I}}^{\boldsymbol{2}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{3}}$. This constraint invokes a precedence on clear times for incidents where current or future solutions of the model may not require transfer to hospital for these incidents. Incidents that have arrived or are en route to hospital (that is, incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{4}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{5}}$) have a clear time determined by time spent at hospital:

$$c_i \geq r_i + x_{ia}(\gamma_{ia} + \lambda_{ia}) \qquad \forall i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{1}}, \hat{\boldsymbol{I}}^{\boldsymbol{2}}, \hat{\boldsymbol{I}}^{\boldsymbol{3}}\}, a \in \boldsymbol{A}, \tag{8.15}$$

Constraint (8.16): Incidents assigned to any hospital cannot be considered clear until the incident has been admitted at the hospital and the ambulance cleaned for a new job:

$$c_i \geq g_i + y_{ih}(\zeta_{ih} + \eta_{ih}) + x_{ia}\lambda_{ia} \qquad \forall i \in \boldsymbol{I}, a \in \boldsymbol{A}, h \in \boldsymbol{H} \tag{8.16}$$

It is not necessary, in the real time model, to have constraint preventing the clear time from changing once it has been met as these jobs will no longer be present in the model.

Meal breaks, return-to-station jobs and relocations require dispatch and clear times to related through precedence constraints.

Constraint (8.17): Clear times for meal breaks must not be earlier than the dispatch time:

$$c_i \geq d_i \qquad \forall\, i \in \{\boldsymbol{J}^{\boldsymbol{B1}}, \boldsymbol{J}^{\boldsymbol{B2}}\} \tag{8.17}$$

Constraint (8.18): Clear times for relocation and return-to-station jobs must be later than the dispatch time plus sufficient travel time:

$$c_i \geq d_i + \sum_{n \in \boldsymbol{N}} \delta_{in}\mu_{nl_i} \qquad \forall\, i \in \{\boldsymbol{J}^{\boldsymbol{R}}, \boldsymbol{J}^{\boldsymbol{S}}\} \tag{8.18}$$

Disjunctive constraints

Paired disjunctive constraints apply to all jobs. These ensure that there is no overlap in processing time between multiple jobs assigned to the same ambulance.

Constraints (8.19) and (8.20): Disjunctive variable z_{jia} reflects appropriately whether incident *i* precedes incident *j*, or vice versa, if they are both assigned ambulance *a*:

$$d_i + 2M(1 - x_{ia}) - c_j + 2M\big(1 - x_{ja}\big) \geq M\big(z_{jia} - 1\big) \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\}, a \in \boldsymbol{A} \tag{8.19}$$

$$d_j + 2M\big(1 - x_{ja}\big) - c_i + 2M(1 - x_{ia}) \geq -Mz_{jia} \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\}, a \in \boldsymbol{A} \tag{8.20}$$

Constraints (8.21) and (8.22): The disjunctive variables for jobs *i* and *j* on ambulance *a* should equal zero if either job is not assigned to ambulance *a*. These are supplements to constraints (8.19) and (8.20) which restrict z_{jia} only if both incidents are on ambulance *a*:

$$z_{jia} \leq x_{ia} \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\}, a \in \boldsymbol{A} \tag{8.21}$$

$$z_{jia} \leq x_{ja} \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\}, a \in \boldsymbol{A} \tag{8.22}$$

Constraint (8.23): This constraint applies to the diagonal of the disjunctive variable which is not constrained by either constraint (8.19) or (8.20). It is a guarantee that the diagonal values of the disjunctive variable will be zero, in accordance with requirement that an incident cannot be cleared before it begins:

$$z_{iia} = 0 \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.23}$$

In addition to disjunctive variables preventing overlap between jobs on the same ambulance, additional disjunctive variables are required to prevent ambulances being in two locations at the same time. An immediate predecessor disjunctive variable for jobs is introduced to create this effect in the model. It is effective because information on job destinations and ambulance assignments already forms part of the model. There must be exactly one immediate predecessor for job *i* if there are any jobs preceding it on the same ambulance, as identified in the disjunctive variable z_{jia}. Conversely, there is exactly one antecedent to job *j* if there are any succeeding jobs on the same ambulance as identified by z_{ija}.

Constraint (8.24): The immediate predecessor for job *i* must be zero if there are no jobs preceding job *i* on the ambulance to which it is assigned:

$$Z_{ij} \leq \sum_{a \in \boldsymbol{A}} z_{ija} \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\} \tag{8.24}$$

Constraint (8.25): There must be at least one immediate predecessor for job *i* if there are any preceding jobs on the assigned ambulance:

$$J_{max} \sum_{j \in \boldsymbol{J}/\{i\}} Z_{ij} \geq \sum_{j \in \boldsymbol{J}\{i\}} \sum_{a \in \boldsymbol{A}} z_{ija} \quad \forall\, i \in \boldsymbol{J} \tag{8.25}$$

Constraint (8.26): There cannot be more than one immediate predecessor for any job *i*:

$$\sum_{j \in \boldsymbol{J}} Z_{ij} \leq 1 \quad \forall\, i \in \boldsymbol{J} \tag{8.26}$$

Constraint (8.27): There cannot be more than one immediate antecedent for any job *i*:

$$\sum_{j \in \boldsymbol{J}} Z_{ji} \leq 1 \quad \forall\, i \in \boldsymbol{J} \tag{8.27}$$

Constraint (8.28): A special precedence relation is applied to all return-to-station jobs through use of the disjunctive variables. This constraint ensures that the last job for an ambulance on every shift will return the ambulance to its home ambulance station:

$$z_{ija} \geq x_{ja} - M(1 - x_{ia}) \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J^S} \setminus \{i\}, a \in \boldsymbol{A}, \tag{8.28}$$

Incident resource assignment

Each incident must be assigned to an ambulance and, if required, to a hospital. These must be of a suitable type in accordance with the input associated with each incident. All other jobs only require assignment to an ambulance.

Constraint (8.29): Each incident or return-to-station job *i* must be assigned to exactly one ambulance:

$$\sum_{a \in \boldsymbol{A}} x_{ia} = 1 \quad \forall\, i \in \{\boldsymbol{I}, \boldsymbol{J^s}\} \tag{8.29}$$

Constraint (8.30): Unique relocation jobs and breaks can occur at most once each time the model is solved. Unique breaks should occur once per shift and disappear from the set of jobs in the parameters if they have been cleared prior to the real time model being initialised:

$$\sum_{a \in \boldsymbol{A}} x_{ia} \leq 1 \quad \forall\, i \in \{\boldsymbol{J^R}, \boldsymbol{J^{B1}}, \boldsymbol{J^{B2}}\} \tag{8.30}$$

Constraint (8.31): The ambulance *a* that is assigned to job *i* must be one of the ambulances identified as able to provide an appropriate response:

$$x_{ia} \leq \xi_{ia} \qquad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.31}$$

Constraint (8.32): Incidents are required by the model to retain the same ambulance once the ambulance has arrived at the scene of the incident. This constraint is required for incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{3}}$, $\hat{\boldsymbol{I}}^{\boldsymbol{4}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{5}}$:

$$x_{ia} \geq \check{x}_{ia} \qquad \forall\, i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{3}}, \hat{\boldsymbol{I}}^{\boldsymbol{4}}, \hat{\boldsymbol{I}}^{\boldsymbol{5}}\}, a \in \boldsymbol{A} \tag{8.32}$$

Incidents where transfer to hospital is required must be directed to a single hospital of a type requested by the parameter for the incident. This requires the following set of constraints.

Constraint (8.33): Any incident *i* can be directed to at most one hospital:

$$\sum_{h \in \boldsymbol{H}} y_{ih} \leq 1 \qquad \forall\, i \in \boldsymbol{I} \tag{8.33}$$

Constraint (8.34): An incident *i* must be directed to at least one hospital if hospital transfer is required:

$$M \sum_{h \in \boldsymbol{H}} y_{ih} \geq \sum_{h \in \boldsymbol{H}} \pi_{ih} \qquad \forall\, i \in \boldsymbol{I} \tag{8.34}$$

Constraint (8.35): Incident *i* can only be sent to hospital *h* if the hospital is appropriate for incident *i*:

$$y_{ih} \leq \pi_{ih} \qquad \forall\, i \in \boldsymbol{I}, h \in \boldsymbol{H} \tag{8.35}$$

Constraint (8.36): Incidents that have already arrived at a hospital (that is, incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{5}}$) are required by the model to keep the same hospital assignment:

$$y_{ih} \geq \check{y}_{ih} \qquad \forall\, i \in \hat{\boldsymbol{I}}^{\boldsymbol{5}}, h \in \boldsymbol{H} \tag{8.36}$$

Location constraints

Location constraints are used to keep track of the position of all ambulances. Ambulance location updates when operations involving travel are completed and are queried each time the real time model is initialised.

Constraint (8.37): Where job *i* is preceded immediately by job *j*, the dispatch location of the job *i* is the same as the clear location of the previous job *j*:

$$\delta_{in} \geq \theta_{jn} + M(Z_{ji} - 1) \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\}, n \in \boldsymbol{N} \tag{8.37}$$

Constraint (8.38): Jobs have exactly one dispatch location if an ambulance is assigned to that job or no dispatch location if no ambulance is assigned:

$$\sum_{n \in \boldsymbol{N}} \delta_{in} = \sum_{a \in \boldsymbol{A}} x_{ia} \quad \forall\, i \in \boldsymbol{J} \tag{8.38}$$

Constraint (8.39): Similarly, each job *i* has exactly one clear location if assigned an ambulance and no clear location otherwise:

$$\sum_{n \in \boldsymbol{N}} \theta_{in} = \sum_{a \in \boldsymbol{A}} x_{ia} \quad \forall\, i \in \boldsymbol{J} \tag{8.39}$$

Constraint (8.40): Where an incident *i* requires transfer to a hospital, it must have exactly one location from which transfer commences:

$$\sum_{n \in \boldsymbol{N}} \Delta_{in} = \sum_{h \in \boldsymbol{H}} y_{ih} \quad \forall\, i \in \boldsymbol{I} \tag{8.40}$$

Constraint (8.41): If a response to an incident *i* has already arrived at the scene of an incident at time *t* (that is, incident status $\hat{\boldsymbol{I}}^{\boldsymbol{3}}$, $\hat{\boldsymbol{I}}^{\boldsymbol{4}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{5}}$) then dispatch location remains the same as in the previous solution:

$$\delta_{in} \geq \check{\delta}_{in} \quad \forall\, i \in \{\hat{\boldsymbol{I}}^{\boldsymbol{3}}, \hat{\boldsymbol{I}}^{\boldsymbol{4}}, \hat{\boldsymbol{I}}^{\boldsymbol{5}}\}, n \in \boldsymbol{N} \tag{8.41}$$

Constraint (8.42): Incidents requiring transfer to a hospital that have not yet completed treatment at the scene of the incident (that is, incidents with status $\hat{\boldsymbol{I}}^{\boldsymbol{1}}$, $\hat{\boldsymbol{I}}^{\boldsymbol{2}}$ or $\hat{\boldsymbol{I}}^{\boldsymbol{3}}$) will always have transfer to any hospital beginning from the location of the scene of the incident:

$$\Delta_{iL_i} \geq \pi_{ih} \quad \forall\, i \in \boldsymbol{I}, h \in \boldsymbol{H} \tag{8.42}$$

Constraint (8.43): Where an incident is still en route to a hospital and no reassignment occurs, the location from which transfer to hospital begins remains the same as in the previous solution:

$$\Delta_{in} \geq \check{\Delta}_{in} - M(2 - y_{ih} - \check{y}_{ih}) \quad \forall\, i \in \hat{\boldsymbol{I}}^{\boldsymbol{4}}, n \in \boldsymbol{N} \tag{8.43}$$

Constraint (8.44): Similarly, where incident *i* has already arrived at hospital *h* at time *t* (that is, incident status $\hat{\boldsymbol{I}}^{\boldsymbol{5}}$) then the location from which transfer to hospital begins remains the same as in the previous solution:

$$\Delta_{in} \geq \breve{\Delta}_{in} \quad \forall\, i \in \hat{\boldsymbol{I}}^{\mathbf{5}}, n \in \boldsymbol{N} \tag{8.44}$$

Constraint (8.45): In the event of a hospital reassignment for incident *i* currently en route to a hospital, the location from which transfer to hospital begins becomes the current location of the ambulance:

$$\Delta_{i\Gamma_a} \geq y_{ih} - \breve{y}_{ih} \quad \forall\, i \in \hat{\boldsymbol{I}}^{\mathbf{4}}, n \in \boldsymbol{N} \tag{8.45}$$

Constraint (8.46): For any incident *i* where transfer to a hospital occurs, the clear location of incident *i* will be the location of the hospital:

$$\theta_{iN_h} \geq y_{ih} \quad \forall\, i \in \boldsymbol{I}, h \in \boldsymbol{H} \tag{8.46}$$

Constraint (8.47): For incidents *i* where there is no transfer to a hospital, the clear location must be the location of incident *i*:

$$\theta_{iL_i} \geq 1 - \sum_{h \in \boldsymbol{H}} y_{ih} \quad \forall\, i \in \boldsymbol{I} \tag{8.47}$$

Constraint (8.48): The clear location for relocation and return-to-station jobs is the station to which an ambulance is directed (that is, the assigned destination L_i) if an ambulance is assigned to the job:

$$\theta_{iL_i}(t) \geq \left(\sum_{a \in \boldsymbol{A}} x_{ia} \right) \quad \forall\, i \in \{\boldsymbol{J}^{\boldsymbol{S}}, \boldsymbol{J}^{\boldsymbol{R}}\} \tag{8.48}$$

Constraint (8.49): Breaks are assumed to clear at the same location where they started each time the real time model is saved. Breaks in progress will be updated with new locations each time the real time model is called:

$$\theta_{in} = \delta_{in} \quad \forall\, i \in \{\boldsymbol{J}^{\boldsymbol{B1}}, \boldsymbol{J}^{\boldsymbol{B2}}\}, n \in \boldsymbol{N} \tag{8.49}$$

Constraint (8.50): If there are no predecessors to job *i*, and the dispatch location is not carried from the previous solution, then the dispatch location becomes the current location of the ambulance at time *t*:

$$\delta_{in} \geq \delta_{i\Gamma_a} - M \sum_{j \in \boldsymbol{J}} Z_{ji} - M(1 - x_{ia}) \quad \forall\, i \in \{\hat{\boldsymbol{I}}^{\mathbf{1}}, \hat{\boldsymbol{I}}^{\mathbf{2}}, \boldsymbol{J}^{\boldsymbol{S}}, \boldsymbol{J}^{\boldsymbol{R}}, \boldsymbol{J}^{\boldsymbol{B1}}, \boldsymbol{J}^{\boldsymbol{B2}}\}, \; a \in \boldsymbol{A} \tag{8.50}$$

Constraint (8.51): An ambulance is not permitted to idle for any length of time at a location other than an ambulance station. This is handled by restricting the amount of time between clearing one job and starting the next to one minute or less whenever the clear location is other than an ambulance station:

$$d_j - c_i - M(1 - Z_{ij}) \leq 1 + M\left(1 - \left(\theta_{in} - \sum_{m \in N_s} \theta_{im}\right)\right) \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\}, \; n \in \boldsymbol{N} \tag{8.51}$$

Tardiness constraints

Expected tardiness for each incident is determined by the arrival time of a response for each incident and the desirable response time to fit the performance measures.

Constraint (8.52): Normal tardiness occurs for arrival times after the first response time limit. Arrival times resulting in excessive tardiness only accrue normal tardiness up until the time when excessive tardiness begins (that is, the second response time limit):

$$\tau_i \geq r_i - T_i - \tau_i' \qquad \forall\, i \in \boldsymbol{I} \tag{8.52}$$

Constraint (8.53): Excessive tardiness occurs for arrival times after the second response time limit:

$$\tau_i' \geq r_i - U_i \qquad \forall\, i \in \boldsymbol{I} \tag{8.53}$$

Overtime constraints

Expected overtime is returned each time the real time model is solved. The value of overtime for ambulance *a* is based on the time it is expected to return to its home ambulance station and the time it is due to complete the current shift.

Constraint (8.54): Overtime accrued by ambulance *a* is greater than or equal to the clear time of job *j* returning ambulance *a* to the correct ambulance station at the end of a shift, minus the time when the ambulance is meant to complete the current shift as per the shift schedule:

$$c_i - E_a - o_a \leq M(1 - x_{ia}) \qquad \forall\, i \in \boldsymbol{J^S}, a \in \boldsymbol{A} \tag{8.54}$$

Look ahead constraints

Look ahead constraints predict the availability and location of each ambulance at time $\hat{t}$. This is used to work out ambulance movements to improve coverage. Dispatching an ambulance to an incident or clearing incidents and return-to-station jobs will affect the availability of ambulances:

Constraints (8.55) & (8.56): These constraints find all jobs assigned to ambulance a that occur prior to the look ahead time $\hat{t}$ in the real time model:

$$M\hat{\chi}_{ia} \geq \hat{t} - d_i - M(1 - x_{ia}) \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.55}$$

$$\hat{\chi}_{ia} \leq x_{ia} \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.56}$$

Constraint (8.57): This constraint uses the variable $\hat{\chi}_{ia}$ controlled in constraints (8.55) and (8.56) and extracts the job dispatched last, prior to time $\hat{t}$. This incident is assigned to the variable $\widehat{\Psi}_{ia}$, determining the last job to occur before the look ahead time for coverage. Both variables are required in order to avoid non-linearity:

$$\widehat{\Psi}_{ia} \geq \hat{\chi}_{ia} - MZ_{ij} - M\left(1 - \hat{\chi}_{ja}\right) \quad \forall\, i \in \boldsymbol{J}, j \in \boldsymbol{J} \setminus \{i\}, a \in \boldsymbol{A} \tag{8.57}$$

Constraint (8.58): If there are any jobs assigned to ambulance a, with a dispatch time prior to $\hat{t}$, then there must be a job that is last to occur before time $\hat{t}$:

$$\sum_{i \in \boldsymbol{J}} M\widehat{\Psi}_{ia} \geq \sum_{i \in \boldsymbol{J}} \hat{\chi}_{ia} \quad \forall\, a \in \boldsymbol{A} \tag{8.58}$$

Constraint (8.59): Each ambulance can only have one event occur as the last event before the look ahead time:

$$\sum_{i \in \boldsymbol{J}} \widehat{\Psi}_{ia} \leq 1 \quad \forall\, a \in \boldsymbol{A} \tag{8.59}$$

Constraint (8.60): The last job to occur before the look ahead time for ambulance a must be one of the jobs identified as occurring before time $\hat{t}$ on ambulance a:

$$\widehat{\Psi}_{ia} \leq \hat{\chi}_{ia} \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.60}$$

The following constraints determine the availability of ambulance a at the look ahead time $\hat{t}$. There are several states than an ambulance may be in which determine availability.

Constraints (8.61): Ambulance a is in a state of 'en route to an incident' ($\widehat{\mathrm{A}}_a^1$) if last job to occur prior to the look ahead time is an incident which has not yet received a response at time $\hat{t}$. An ambulance in state $\widehat{\mathrm{A}}_a^1$ is available for reassignment:

$$-M\left(1 - \widehat{\Psi}_{ia} + \widehat{\mathrm{A}}_a^1\right) \leq \hat{t} - r_i \quad \forall\, i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{8.61}$$

Constraint (8.62): Ambulance a may be in a state of either 'busy' ($\widehat{\mathrm{A}}_a^2$) or 'cleared' ($\widehat{\mathrm{A}}_a^3$) if incident i, which was last job to occur prior to time $\hat{t}$, received a

response prior to the look ahead time. An ambulance in state $\widehat{\mathrm{A}}_a^2$ is unavailable. An ambulance in state $\widehat{\mathrm{A}}_a^3$ is available:

$$-M\left(1-\widehat{\Psi}_{ia}+\widehat{\mathrm{A}}_a^2+\widehat{\mathrm{A}}_a^3\right) \leq r_i - \hat{t} \quad \forall\, i \in \boldsymbol{I}, a \in \boldsymbol{A} \tag{8.62}$$

Constraint (8.63): Ambulance *a* cannot be in a state of 'busy' if the last job to occur prior to time $\hat{t}$ was not an incident:

$$\widehat{\Psi}_{ia} + \widehat{\mathrm{A}}_a^2 \leq 1 \quad \forall\, i \in \boldsymbol{J} \setminus \boldsymbol{I}, a \in \boldsymbol{A} \tag{8.63}$$

Constraint (8.64): Ambulance *a* is either in state 'en route' or 'busy' if the last job to occur prior to time $\hat{t}$ received a response but has not yet cleared:

$$-M\left(1-\widehat{\Psi}_{ia}+\widehat{\mathrm{A}}_a^1+\widehat{\mathrm{A}}_a^2\right) \leq \hat{t} - c_i \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.64}$$

Constraint (8.65): Ambulance *a* is either in state 'cleared' or 'ended shift' ($\widehat{\mathrm{A}}_a^6$) if the last job to occur prior to time $\hat{t}$ has cleared:

$$-M\left(1-\widehat{\Psi}_{ia}+\widehat{\mathrm{A}}_a^3+\widehat{\mathrm{A}}_a^6\right) \leq c_i - \hat{t} \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.65}$$

Constraint (8.66): If any job occurs prior to time $\hat{t}$, then an ambulance *a* must be in a state of 'en route', 'busy', 'cleared' or 'ended shift':

$$\widehat{\mathrm{A}}_a^1 + \widehat{\mathrm{A}}_a^2 + \widehat{\mathrm{A}}_a^3 + \widehat{\mathrm{A}}_a^6 = \sum_{i \in \boldsymbol{J}} \widehat{\Psi}_{ia} \quad \forall\, a \in \boldsymbol{A} \tag{8.66}$$

Constraint (8.67): Ambulance *a* must be in any state except 'ended shift' if the look ahead time is prior to the time at which the current shift is due to end:

$$-M\left(1-\widehat{\Psi}_{ia}+\widehat{\mathrm{A}}_a^1+\widehat{\mathrm{A}}_a^2+\widehat{\mathrm{A}}_a^3+\widehat{\mathrm{A}}_a^4+\widehat{\mathrm{A}}_a^5\right) \geq E_a - \hat{t} \quad \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A} \tag{8.67}$$

Constraint (8.68): Ambulance *a* must be in either state 'not yet dispatched' ($\widehat{\mathrm{A}}_a^4$) or 'not yet commenced shift' ($\widehat{\mathrm{A}}_a^4$) if no jobs are found to occur prior to time $\hat{t}$:

$$\widehat{\mathrm{A}}_a^4 + \widehat{\mathrm{A}}_a^5 = 1 - \sum_{i \in \boldsymbol{J}} \widehat{\Psi}_{ia} \quad \forall\, a \in \boldsymbol{A} \tag{8.68}$$

Constraint (8.69): An ambulance is in a state of 'not yet commenced shift' if the look ahead time is later than the time at which the current shift is due to begin:

$$M\left(\widehat{\mathrm{A}}_a^5\right) \geq B_a - \hat{t} \quad \forall\, a \in \boldsymbol{A} \tag{8.69}$$

Coverage constraints

Coverage variables ρ'_n and $\hat{\rho}_{an}$ are used to make informed decisions on relocations. Binary variable $\hat{\rho}_{an}$ describes the nodes *n* in the model which are expected to be covered by ambulance *a* at the look ahead time, $\hat{t}$, which occurs a short period of time after the model is solved. Decision variable ρ'_n indicates whether node *n* is sufficiently covered at the look ahead time.

Constraint (8.70): Ambulance *a* cannot cover node *n* if it is busy or ended its shift prior to time $\hat{t}$:

$$\hat{\rho}_{an} \leq 1 - \hat{A}_a^2 - \hat{A}_a^5 - \hat{A}_a^6 \quad \forall\, a \in \boldsymbol{A}, n \in \boldsymbol{N_G} \tag{8.70}$$

Constraint (8.71): An ambulance *a* in state 'en route' can contribute to coverage of node *n* if the travel time from the last known location of ambulance *a* to node *n* is less than eight minutes:

$$M(1 - \hat{\rho}_{an}) \geq -8 + \sum_{m \in \boldsymbol{N}} \mu_{mn}\delta_{im} - M\left(1 - \hat{\psi}_{ia}\right) - M(1 - A_a^1) \quad \begin{matrix} \forall\, i \in \boldsymbol{J}, \\ a \in \boldsymbol{A}, \\ n \in \boldsymbol{N_G} \end{matrix} \tag{8.71}$$

Constraint (8.72): An ambulance *a* in state 'cleared' can contribute to coverage of node *n* if the travel time from the last known location of ambulance *a* to node *n* is less than eight minutes:

$$M(1 - \hat{\rho}_{an}) \geq -8 + \sum_{m \in \boldsymbol{N}} \mu_{mn}\theta_{im} - M\left(1 - \hat{\psi}_{ia}\right) - M(1 - A_a^3) \quad \begin{matrix} \forall\, i \in \boldsymbol{J}, \\ a \in \boldsymbol{A}, \\ n \in \boldsymbol{N_G} \end{matrix} \tag{8.72}$$

Constraint (8.73): An ambulance *a* in state 'not yet dispatched' can contribute to coverage of node *n* if the travel time from the last known location of ambulance *a* to node *n* is less than eight minutes:

$$M(1 - \hat{\rho}_{an}) \geq -8 + \mu_{\Gamma_a n} - M\left(1 - \hat{\psi}_{ia}\right) - M(1 - A_a^4) \quad \begin{matrix} \forall\, a \in \boldsymbol{A}, \\ n \in \boldsymbol{N_G} \end{matrix} \tag{8.73}$$

Constraint (8.74): The coverage gap variable must be positive for node *n* if the number of ambulances covering node *n* is fewer than the number requested in the coverage requirements parameter. For this constraint to avoid singularities, the coverage requirements may not be equal to zero:

$$\rho'_n \geq 1 - \sum_{a \in \boldsymbol{A}} \hat{\rho}_{an} / \rho_n \quad \forall\, n \in \boldsymbol{N_G} \tag{8.74}$$

Break constraints

The objective function contains binary variables specifying whether penalties for untaken or interrupted meal breaks apply for each ambulance *a*. These constraints are noted here.

Constraint (8.75): If a meal break is not taken at all for any ambulance *a* then apply the appropriate penalty for the ambulance lacking a meal break. A mealbreak

is defined to be untaken if the clear time for the break is less than 50% of the processing time:

$$M\,\hat{\beta}_a^1 \geq d_i + 0.5 \times \gamma_{ia} - c_i + M(1 - x_{ia}) \qquad \forall i \in \boldsymbol{J^{B1}}, a \in \boldsymbol{A}\ s.t.\ \xi_{ia} = 1 \tag{8.75}$$

Constraint (8.76): If a rest break is not taken at all for any ambulance *a* then apply the appropriate penalty for the ambulance lacking a rest break. A rest break is defined to be untaken if the clear time for the break is less than 50% of the processing time:

$$M\,\hat{\beta}_a^2 \geq d_i + 0.5 \times \gamma_{ia} - c_i + M(1 - x_{ia}) \qquad \forall i \in \boldsymbol{J^{B2}}, a \in \boldsymbol{A}\ s.t.\ \xi_{ia} = 1 \tag{8.76}$$

Constraint (8.77): Where an ambulance *a* is allocated a meal break of duration between 50% and 100% of the required meal break time, a penalty for an interrupted meal break is applied to ambulance *a*:

$$M\,\beta_a^1 \geq -M\,\hat{\beta}_a^1 + d_i + \gamma_{ia} - c_i - M(1 - x_{ia}) \qquad \forall i \in \boldsymbol{J^{B1}}, a \in \boldsymbol{A} \tag{8.77}$$

Constraint (8.78): Where an ambulance *a* is allocated a rest break of duration between 50% and 100% of the required rest break time, a penalty for an interrupted rest break is applied to ambulance *a*:

$$M\,\beta_a^2 \geq -M\,\hat{\beta}_a^2 + d_i + \gamma_{ia} - c_i - M(1 - x_{ia}) \quad \forall i \in \boldsymbol{J^{B2}}, a \in \boldsymbol{A} \tag{8.78}$$

Constraints (8.79) and (8.80): Meal breaks must be taken within the appropriate time window. If a meal break is completed but not taken within the required time window, the penalty for an interrupted meal is imposed. This penalty is applied for either an interrupted meal break or for a meal break outside the time window. In the event that the meal break is both interrupted and outside the time window, the penalty is only applied once:

$$M\,\beta_a^1 \geq -M\,\hat{\beta}_a^1 + (B_a + 4 \times 60) - d_i \qquad \forall i \in \boldsymbol{J^{B1}}, a \in \boldsymbol{A} \tag{8.79}$$

$$M\,\beta_a^1 \geq -M\,\hat{\beta}_a^1 + d_i - M\big(1 - x_{ia}(t)\big) - (B_a + 6 \times 60) \qquad \forall i \in \boldsymbol{J^{B1}},\ a \in \boldsymbol{A} \tag{8.80}$$

Relocation constraints

Constraint (8.81): Where multiple relocation jobs with the same destination are introduced, utilise the appropriate jobs with the lowest index first. This constraint enforces a particular solution where duplicate solutions exist:

$$\sum_{a \in \boldsymbol{A}} x_{ia} \leq \sum_{a \in \boldsymbol{A}} x_{ja} \qquad \forall\, i \in \boldsymbol{J^R}, j \in \boldsymbol{J^R} : (j > i \;\&\; L_j = L_i) \tag{8.81}$$

Constraint (8.82): Relocation jobs from one location to the same location are to be prevented:

$$\theta_{in}(t) + \delta_{in}(t) \leq 1 \qquad \forall\, i \in \boldsymbol{J^R}, n \in \boldsymbol{N} \tag{8.82}$$

Non negativity and integer constraints

Constraint (8.83): All time stamps in the model should be greater than zero and less than the largest time value in the model:

$$0 \leq d_i, r_i, e_i, g_i, c_i \leq M \qquad \forall\, i \in \boldsymbol{J} \tag{8.83}$$

Constraint (8.84): Overtime should also be greater than zero and less than the largest time value in the model:

$$0 \leq o_a \leq M \qquad a \in \boldsymbol{A} \tag{8.84}$$

Constraint (8.85): Dispatch and clear locations can only take on the values of the nodes specified in the model:

$$0 \leq \delta_{in}, \Delta_{in}, \theta_{in} \leq N_{max} \quad \forall\, i \in \boldsymbol{J}, n \in \boldsymbol{N} \tag{8.85}$$

Constraint (8.86): The following dependent variables should be binary:

$$\begin{matrix} x_{ia},\ y_{ih},\ z_{ija},\ Z_{ij},\ \widehat{\Psi}_{ia},\ \hat{\chi}_{ia},\ \widehat{\rho}_{an}, \\ \hat{A}_a^{\hat{a} \in \{1,2,3,4,5,6\}},\ \beta_a^{m \in \{1,2\}},\ \widehat{\beta}_a^{m \in \{1,2\}} \end{matrix} \quad \in \{0,1\} \quad \begin{matrix} \forall\, i \in \boldsymbol{J}, a \in \boldsymbol{A}, \\ h \in \boldsymbol{H}, n \in \boldsymbol{N} \end{matrix} \tag{8.86}$$

Constraint (8.87): The decision variable for coverage gaps has maximum and minimum values defined for each node:

$$0 \leq \rho'_n \leq 1 \qquad \forall\, n \in \boldsymbol{N} \tag{8.87}$$

Constraint (8.88): The decision variable for normal tardiness is non-negative and limited to a maximum value dependent on the response time window for each incident:

$$0 \leq \tau_i \leq U_i - T_i \quad \forall\, i \in \boldsymbol{I} \tag{8.88}$$

Constraint (8.89): The decision variable for excessive tardiness is non-negative and less than the largest time value in the model:

$$0 \leq \tau'_i \leq M \qquad \forall\, i \in \boldsymbol{I} \tag{8.89}$$

8.3 SOLUTION APPROACH

This section discusses the required data and issues for solving the real time model. The formulation allows the model to be solved at any time *t*. The time at which the model is solved may be a trigger point (for example, at the time a new incident arrives) to react to a new system state or at time intervals (for example, every five minutes) to update the systems.

8.3.1 Real Time Information

The model requests real time information every time it is triggered. The information requested includes: ambulance location and availability; incident requirements and progress; expected travel times; expected times for treatment required at the scene of an incident; and expected ramping times at hospitals. This information is expected to be updated externally to the real time model to provide the most accurate and up to date information. Ambulance availability is also related to ambulance crew shift schedules. A section of a pre-planned ambulance crew schedule containing any ambulance crew: scheduled to work a shift at time *t*; utilising overtime at time *t*; or due to become active in the near future; is included in the requested information.

The case study used in the static and dynamic models is insufficient to provide this information. This is for several reasons. The first is because the ambulance location information contained in the case study is limited to hospitals, ambulance stations and incident scenes. The mathematical model only updates ambulance location internally when an operation is completed and the ambulance does one of the following: arrives at scene of an incident; arrives at a hospital; or arrives at an ambulance station. The real time model can be triggered at times in between these events occurring and requires updated locations to be determined externally. The second reason that the previously used case study is insufficient is because one of the benefits of the real time model, updating incident requirements or ambulance assignments in response to changing conditions, will not be explored with a data set where these are stable. The case study maintains the same incident requirements, travel times and processing times for operations.

Addressing these issues would require significant modifications to the case study. Integrating a road network for estimating locations of ambulances en route to

an assigned destination and for updating travel times to all locations is suggested for dealing with the lack of ambulance location information between operations and changing travel times. Allowing perturbations to occur in other processing times (for example, treatment times at incident scenes and ramping times at hospitals) each time the real time model is triggered would allow for exploration of the effects of time varying processing times on ambulance assignments. However, care must be taken when perturbing processing times to maintain realistic variations and distributions for processing times. Integrating a road network is beyond the scope of this project, and information available does not describe how much estimates of processing time might vary for the same incident over a short time period. Instead of modifying the case study, a sample of results from the dynamic model is used to verify the model. Collection of a new, real time data set for further testing of the model is recommended for future work. A real time data set should allow for incident escalation, changing road conditions and updates to hospital ramping times from changing estimates of ED capacity.

8.3.2 Problem Size

Fast solutions for realistic problems are required for the real time model to be useful for decision makers. This section discusses the expected size of a realistic problem and explains whether heuristic solution techniques are expected to be necessary for solving the real time model.

The number of variables for the real time model can be estimated as follows. Assume a very small case study with five ambulances, five ambulance stations, two hospitals and a 4x4 coverage grid. At the beginning of a shift, each ambulance requires two breaks. Each ambulance must also be returned to the correct ambulance station at the end of a shift. An ambulance may be relocated after responding to an incident. This relocation may direct an ambulance to any ambulance station within the case study. Relocations may also occur to improve coverage each time the real time model is solved. All relocation jobs to which ambulances may potentially be assigned should be present in the model. The number of potential relocations is estimated at $I_{max} \times S_{max} + A_{max} \times S_{max}$. The total number of jobs in the model, containing two breaks per ambulance, one return-to-station job per ambulance, all potential relocations and all incidents, is estimated to be: $2 \times A_{max} + A_{max} + (I_{max} \times S_{max} + A_{max} \times S_{max}) + I_{max} = A_{max}(S_{max} + 3) + I_{max}(S_{max} + 1)$.

Output from the dynamic model showed that the number of active incidents at a time could easily be 20 incidents during peak demand, resulting in 360 potential relocations for the small example problem. This estimation of jobs in a small size model is used to estimate the number of variables the real time model may encounter. In reality, the number of ambulances, ambulance stations, hospitals and nodes in the system would be much greater for the entire metropolitan region. Realistic problems would be much larger than this example.

Figure 8-3 shows an estimation of the total number of variables present in the model for increasing numbers of incidents, ambulances or stations. It can be seen that the number of variables increases significantly when these parameters increase, fuelled by an increasing number of disjunctive variables. The number of relocation jobs (fuelled by incidents, ambulances and stations) is the main driver for the size of the problem.

The total number of variables could be reduced slightly by attempting to reduce the number of possible relocation jobs considered in the model. Not all of the relocation jobs are expected to be selected in a schedule but they must be available as it is not known which relocations form part of the best schedule. Firstly, only ambulances that are suitable to respond to an incident are able to be relocated after that incident is cleared. Secondly, relocations that are unlikely to occur could be eliminated from the model. This would reduce the number of disjunctive variables and the total size of the problem.

However, there are still large numbers of variables that cannot be reduced. Real time solutions for this model are expected to require a heuristic approach in order to return solutions quickly enough to be useful to a decision maker.

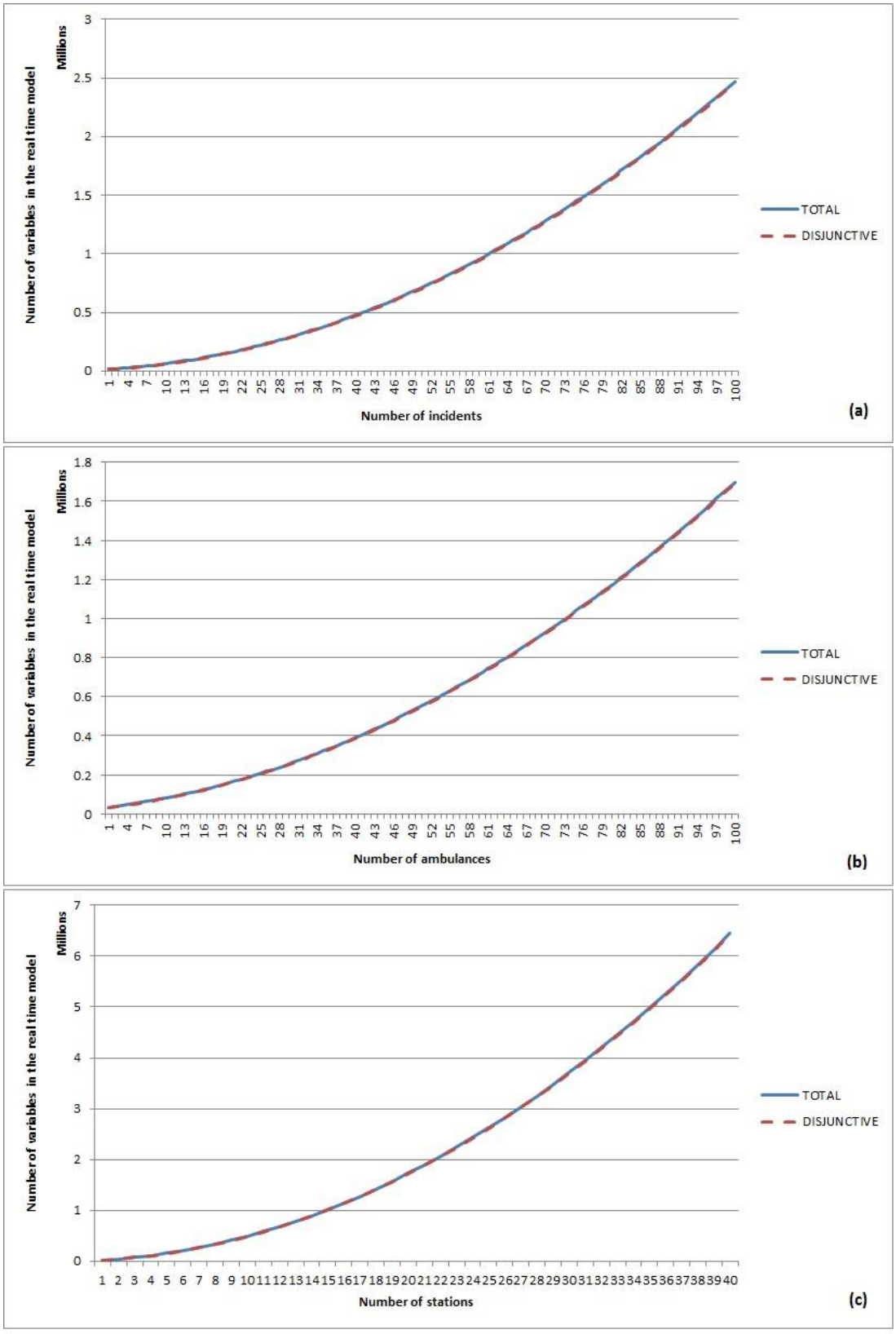


Figure 8-3 Estimated number of variables in the real time model for a sample scenario

8.3.3 Case Study

Due to the expected size of the problem, a single, simple scenario is developed for validating the real time model. A scenario, using realistic demand from the case study in Chapter 5 and results from the best schedule in Chapter 7, is extracted. The case study is initialised during a period of off-peak demand to keep the size of the case study small. A small case study allows the real time model to be solved using a MIP solver. The best shift schedule found from the dynamic model is to be used to initialise ambulance shifts and ambulance station assignments in the real time model. Operations scheduled to occur prior to the time when the case study begins, as found from the schedule in the dynamic model, are fixed events in the case study for the real time model.

The real time model is triggered by the arrival of a new incident in the case study at t = 1721, equivalent to 4:41 am on a Saturday morning. This period of time only contains a single shift of ambulances and is past the time window for meal breaks. For the case study, it is assumed that all meal breaks were able to be met within the time window but a number of rest breaks remain to be scheduled. Ten ambulances were scheduled amongst the five ambulance stations in the case study for the shift covering time t = 1721. Each of these requires a return-to-station job. It was arbitrarily decided to schedule rest breaks for four of these ten ambulances and assume that the other four have already met the requirement for a scheduled rest break. Potential relocation jobs are inserted into the model allowing ambulances to be relocated to other ambulance stations in order to improve coverage. At time t = 1721, there are two incidents already in progress. Both of these incidents are in progress at hospitals. The only incident waiting for an ambulance is the incident which arrived at time t. The case study was able to be limited to <30 jobs (including all incidents, returns to station, rest breaks and relocations). Tables 8-1 and 8-2 show the jobs (with a selection of parameters) and ambulances present in the real time model when it is triggered by the case study at time t = 1721.

These tables show a number of ambulances available at ambulance stations. Some of these are already at their home ambulance station, so that a return-to-station job will have a travel time of zero, and others are located elsewhere as a result of previous relocations. One ambulance is in the middle of relocating at the time when the real time model is triggered. A new location node, with associated new travel

Table 8-1 Case study job data for real time model triggered at time t = 1721

i	*Job Type*	R_i	T_i	U_i	L_i	*Suitable ambulances*	*Existing assignments*						
							$a: \check{x}_{ia} = 1$	$h: \check{y}_{ih} = 1$	$\check{d}_i$	$\check{r}_i$	$\check{e}_{ih}: \check{y}_{ih} = 1$	$\check{g}_{ih}: \check{y}_{ih} = 1$	$\check{c}_i$
1	Incident (P2)	1663	1693	1723	8	5	5	1	1663.00	1666.57	1671.98	1673.56	1798.42
2	Incident (P2)	1676	1706	1736	9	6	6	1	1676.00	1681.19	1709.06	1716.14	1737.47
3	Incident (P2)	1721	1751	1781	10	{1,2,4,5,6, 7,8,9,10}							
4	Return-to-station	1260			3	3							
5	Return-to-station	1260			3	1							
6	Return-to-station	1260			3	2							
7	Return-to-station	1260			5	4							
8	Return-to-station	1260			3	5							
9	Return-to-station	1260			3	6							
10	Return-to-station	1260			2	7							
11	Return-to-station	1260			5	8							
12	Return-to-station	1260			5	9							
13	Return-to-station	1260			5	10							
14	Rest break	1260				3							
15	Rest break	1260				1							
16	Rest break	1260				4							
17	Rest break	1260				5							
18	Rest break	1260				7							
19	Rest break	1260				8							
20	Relocation	1260			1	any							
21	Relocation	1260			1	any							
22	Relocation	1260			2	any							
23	Relocation	1260			2	any							
24	Relocation	1260			3	any							
25	Relocation	1260			3	any							
26	Relocation	1260			4	any							
27	Relocation	1260			4	any							
28	Relocation	1260			5	any							
29	Relocation	1260			5	any							

Table 8-2 Case study ambulance data for real time model triggered at time $t = 1721$

Ambulance	Shift Start	Shift End	Type	Home Ambulance Station	Location at time $t = 1721$	Status at time $t = 1721$
1	1260	1860	2	3	5	Available at station
2	1260	1860	2	3	3	Available at station
3	1260	1860	3	3	11	Available on route to station
4	1260	1860	2	5	3	Available at station
5	1260	1860	1	3	6	Available at station
6	1260	1860	1	3	6	Busy
7	1260	1860	1	2	5	Busy
8	1260	1860	1	5	5	Available at station
9	1260	1860	1	5	5	Available at station
10	1260	1860	1	5	5	Available at station

times, is generated and included in the model in order to allow this ambulance to be dispatched or relocated from its current position.

Solving the case study triggers the real time model for a single instance. The disadvantage of this approach is that a single instance does not allow analysis of the flow-on effects from decisions made in the schedule to be investigated at future points in time. Real time data is required, containing incident requirements, ambulance locations and travel times to be updated as decisions are acted upon and new information becomes available. The model is solved with the case study to verify that the model works as expected and allows analysis of the multiple criteria objective function. Further work is required to test the response of the model to real time information.

The case study is solved using CPLEX. The model is solved using the multiple criteria objective and for each decomposed component of the objective function.

8.4 RESULTS AND DISCUSSION

The case study was first solved with all components of the objective included in the real time model. The weights for each component in the objective function are shown in Table 8-3. These weights were selected to balance criteria that are measured in units of time, with the potential to be large, against criteria where the units result in smaller values. For the real time model, the units are time for overtime and tardiness criteria, the number of nodes not covered for coverage gaps and the number of breaks where requirements were not met for break penalties. A general understanding of the importance of each criterion has also been applied when selecting weight values. For example, it is more important to have timely response to an emergency incident than it is to reduce overtime. The overtime weights are consistent with the earlier models in Chapters 6 and 7 when selecting values for costs of different ambulance types. The relationship between weights of each of the criteria in the objective function is sensible and expected to be the correct order of magnitude to make sure all items are considered in order of importance. However, further testing with a larger set of real time data is required and refine the values for the objective weights.

A short solution time limit of five minutes was applied to test the quality of solutions that could be gathered quickly. Results obtained from application of all of

the objective criteria, together and independently, are shown in Table 8-4. This table presents the resulting value for each component of the objective function, the solution time and the MIP gap. The response time for the third incident, whose arrival is the trigger for solving the real time model, is also shown. Where decomposed objective criteria were investigated, the resulting schedule was analysed to extract the correct values for other criteria.

8.4.1 Decomposition of objective criteria

It was found that the model was able to be solved to an optimal solution within the 5 minute time limit set for the real time model. Optimal solutions for the case study, for decomposed tardiness, overtime, coverage and break penalties in the objective function, were able to be found in seconds. From decomposed bi-criteria objective solutions (Scenarios 6 through to 11), it can be inferred that overtime and break penalties are competing objectives with each other as overtime does not reach its lowest possible value in an optimal schedule where break penalties form part of the objective. Meal and rest breaks also compete with coverage. While the results from testing Scenario 8 show no drop in coverage from minimising break penalties alone compared to a multiple objective criteria, a thorough analysis of the schedule showed a drop in the number of ambulances available at the look ahead time. Areas that are covered may be covered by only one ambulance. In scenarios with either higher demand or fewer resources, it may not be possible to optimise both break penalties and coverage.

However, the choice not to include response time in the objective function, in favour of minimising tardiness, resulted in higher than necessary response times for the third incident. The impact of including response times in the objective is investigated, with results shown in Table 8-5. The first entry in the table shows the result where response time is given a large weight ($w(r_i)$ = 10000) so that it will be the dominant criteria for the real time model.

This approach is able to return a solution for the case study in less than a minute, with a response time of 4.46 minutes for the new incident. However, the overpowering dominance of the response time criteria resulted in poorer outcomes for almost all of the other criteria (coverage gaps appeared, overtime was greater than in the solution without response time and a break penalty was applied.

Table 8-3 Objective criteria weights for the real time model

Objective criterion	τ_i			τ_i'			ρ_n'	o_a			β_a^2	$\hat{\beta}_a^2$
Weight	ω_1	ω_2	ω_3	ω_1'	ω_2'	ω_3'	κ	σ_1	σ_2	σ_3	Φ^2	$\widehat{\Phi}^2$
	0.15	0.1	0.05	0.03	0.02	0.01	1	0.003	0.002	0.001	0.2	0.1

Table 8-4 Results from the real time model with decomposed objective function for the case study

Scenario	Objective Criteria				Result					
	(τ_i, τ_i')	ρ_n'	o_a	$(\beta_a^2, \hat{\beta}_a^2)$	$r_3 - R_3$ (mins)	$\sum_{i \in I} \tau_i + \tau_i'$	$\sum_{n \in N_G} \rho_n'$	$\sum_{a \in A} o_a$	$\sum_{a \in A} \beta_a^2 + \hat{\beta}_a^2$	CPU time (s)
1	✓	✓	✓	✓	18.59	0	0	76.70	0	254.44
2	✓				19.46	0	0	150.87	6	1.44
3		✓			4.46	0	0	62.57	6	71.35
4			✓		4.46	0	0	62.57	6	3.12
5				✓	19.46	0	0	150.87	0	2.03
6	✓	✓			4.46	0	0	62.57	6	9.36
7	✓		✓		4.46	0	0	62.57	6	2.29
8	✓			✓	19.46	0	0	150.87	0	1.97
9		✓	✓		4.46	0	0	62.57	6	74.52
10		✓		✓	22.10	0	0	100.21	0	19.58
11			✓	✓	18.59	0	0	76.70	0	76.81

Table 8-5 Results from the real time model including response time in the objective function and varying time limit

Scenario	Objective Criteria						Results				
	$w(r_i)$	(τ_i, τ_i')	ρ_n'	o_a	$(\beta_a^2, \hat{\beta}_a^2)$	$r_3 - R_3$ (mins)	$\sum_{i \in I} \tau_i + \tau_i'$	$\sum_{n \in N_G} \rho_n'$	$\sum_{a \in A} o_a$	$\sum_{a \in A} \beta_a^2 + \hat{\beta}_a^2$	CPU time (s)
12	10,000	✓	✓	✓	✓	4.46	0	3	135.87	1	56.57
13	1	✓	✓	✓	✓	4.46	0	0	82.57	0	144.00

However, the choice not to include response time in the objective function, in favour of minimising tardiness, resulted in higher than necessary response times for the third incident. The impact of including response times in the objective is investigated, with results shown in Table 8-5. The first entry in the table shows the result where response time is given a large weight ($w(r_i)$ = 10000) so that it will be the dominant criteria for the real time model.

8.4.2 Sensitivity of look ahead time

The look ahead time parameter has been set to 15 minutes for the case studies tested above. This value is reasonable for look ahead time; it allows sufficient time for the model to be solved and ambulances to make short term relocations for optimal coverage at time $t+\hat{t}$. To verify this, other look ahead times (both shorter and longer) are investigated with results presented in Table 8-6. No difference in optimal solution was found by varying look ahead time from 0 to 60 minutes. There was a slight difference in solution time but no discernible trend. Solution with a look ahead time of 60 minutes took the longest time to solve – this is likely due to an ambulance previously busy with the second incident becoming available again at this point and introducing extra symmetry into the model.

This analysis shows no evidence for selecting one value for look ahead time over another. Users of a real time decision model for ambulance scheduling may wish to select their own look ahead time based on individual operational requirements. It is even possible to vary the coverage parameter each time the model is initialised. In this thesis we have selected a reasonable value of 15 minutes, based on the time to solve the model plus reasonable time for relocation activities to take place prior to the look ahead time, and found it sufficient.

8.4.3 Results Summary

The case study is a single scenario during a period of off-peak demand. There are sufficient resources available to respond to each incident without excessive delays. However, competition still exists between the criteria in the objective. For example, the amount of overtime in an optimised schedule with overtime as the only objective criteria is lower than in a schedule optimised for both overtime and rest breaks. Response time should be included in the objective function to ensure that

Table 8-6 Results from the real time model with varied look ahead time and all objective criteria including response time

Look ahead time (mins)	$r_3 - R_3$ (mins)	$\sum_{i \in \boldsymbol{I}} \tau_i + \tau_i'$	$\sum_{n \in \boldsymbol{N_G}} \rho_n'$	$\sum_{a \in \boldsymbol{A}} o_a$	$\sum_{a \in \boldsymbol{A}} \beta_a^2 + \hat{\beta}_a^2$	CPU time (s)
0	4.46	0	0	82.57	0	101.26
5	4.46	0	0	82.57	0	113.30
10	4.46	0	0	82.57	0	84.29
15	4.46	0	0	82.57	0	144.00
30	4.46	0	0	82.57	0	105.30
45	4.46	0	0	82.57	0	147.62
60	4.46	0	0	82.57	0	164.85

competition between other objectives does not result in delayed responses for patients requiring ambulance services.

Competition between objective criteria is expected to increase in a situation where the utilisation rate of ambulances is higher. A case study during peak demand is suggested as the next step for testing the real time model, prior to testing with real time data. A peak demand case study will be larger and more difficult to solve than the off-peak demand case study. This is due to the additional incidents and ambulances within the system. The added complexity of a peak demand case study is more likely to require heuristic solution techniques to be able to acquire good solutions in a reasonable amount of time.

8.5 HEURISTIC SOLUTION APPROACHES

Heuristic algorithms are proposed as a method of solving the real time model quickly for realistically sized problems. Constructive heuristics were able to solve the dynamic ambulance scheduling model quickly. Hybridisation of the CH with Ant Colony Optimisation and Tabu Search also had promising results. There is potential to adapt the heuristics from Chapter 7 to be suitable for the real time model. However, these require testing and tuning. Further case studies, with appropriate real time data, are needed. Real time data would allow the solutions to be validated across a long period of time, which a single trigger case study cannot do.

8.5.1 Constructive Heuristic

A constructive heuristic, based on the methodology used for the CH in Chapters 6 and 7, has the potential to solve the real time model quickly.

The process diagram for the proposed CH is shown in three parts in Figure 8-4, Figure 8-5 and Figure 8-6. The main part of the CH (seen in Figure 8-4) loops over each job i in the system until each has been scheduled on an ambulance. The first step within the loop is to determine the set of ambulances suitable to be assigned to job i. From there, the heuristic will run a different section of code depending on the type of job. Return-to-station jobs are scheduled to occur last on the ambulance and, if a break is immediately prior, are allowed to pre-empt the break in order to end the shift. Meal and rest break jobs can only be scheduled on a single ambulance but may occur at any time during the shift. Once each required job is scheduled, additional relocation jobs may be added into the schedule if they improve coverage.

The algorithm for assigning incidents is shown in Figure 8-5 (if the incident is waiting for an ambulance to arrive on scene) and Figure 8-6 (if the incident has already received a response).

Incidents waiting for an initial response can be assigned using the process developed for the dynamic model in Chapter 7. First, all potential ambulance assignments are identified and any jobs already scheduled on those ambulances are investigated. These jobs may be predecessors, antecedents or both for the current incident. Hospital options for the incident are also identified. This information determines all paths along which an ambulance responding to the incident might travel (that is, from previous location, to incident scene, to hospital and to next location). Each path is then scored on response times, makespan and the contribution coverage gaps. If either the previously assigned job is a break (meal break or rest break) or the next assigned job is a break or return-to-station, the model allows these to be moved if necessary to avoid overlap between incidents. A penalty score is then added to the path for interrupting breaks or creating more overtime. Paths where overlap exists between incidents are invalid and are scored accordingly. A path is then selected from the options with probability of selection based on the score of each path.

Incidents that have received a response prior to the time at which the model is initialised will have some decision variables already fixed which cannot be changed. Processing times for these incidents will be updated with new real time information. Where the incident has not yet arrived at a hospital, but is required to do so, the algorithm investigates potential paths. Where the updated information creates an overlap between the current job and another job that has been scheduled to begin after the current job, the conflict requires resolving. If the job is a break or return-to-station job that can be moved, the assignment is feasible but will incur a penalty when it is evaluated. If there is overlap between the current incident and another incident, the one with the later dispatching time will be re-scheduled.

8.5.2 Hybrid Heuristics

Previously, it was found that varying the order in which the CH selected incidents for sequencing (through a TS or ACO process) improved solutions when compared with the CH alone. A similar process might also be applied for heuristics

for the real time model. Parameters in the heuristics will require tuning to ensure that solutions are returned quickly enough to be of use to decision makers.

8.6 IMPLICATIONS AND SUMMARY

A real time model has been formulated that can utilise real time information from ambulance services. This model uses an existing ambulance crew shift schedule and searches the solution space for an ambulance schedule suitable for the current situation. Meal and rest breaks are also included in the ambulance schedule. Coverage requirements at a look ahead time are used to relocate ambulances.

As no real time information for ambulance services was available, a case study with a single trigger for the real time model was solved. This was able to verify that the scheduling model can represent real life dispatch and scheduling of ambulances with few simplifying assumptions. Solving the case study also highlighted that including coverage in the objective function made finding an optimal solution much more difficult. This suggests that the approach used in the dynamic model in Chapter 7, where expected incident locations was used to inform ambulance relocation instead of coverage, is worth pursuing in greater detail.

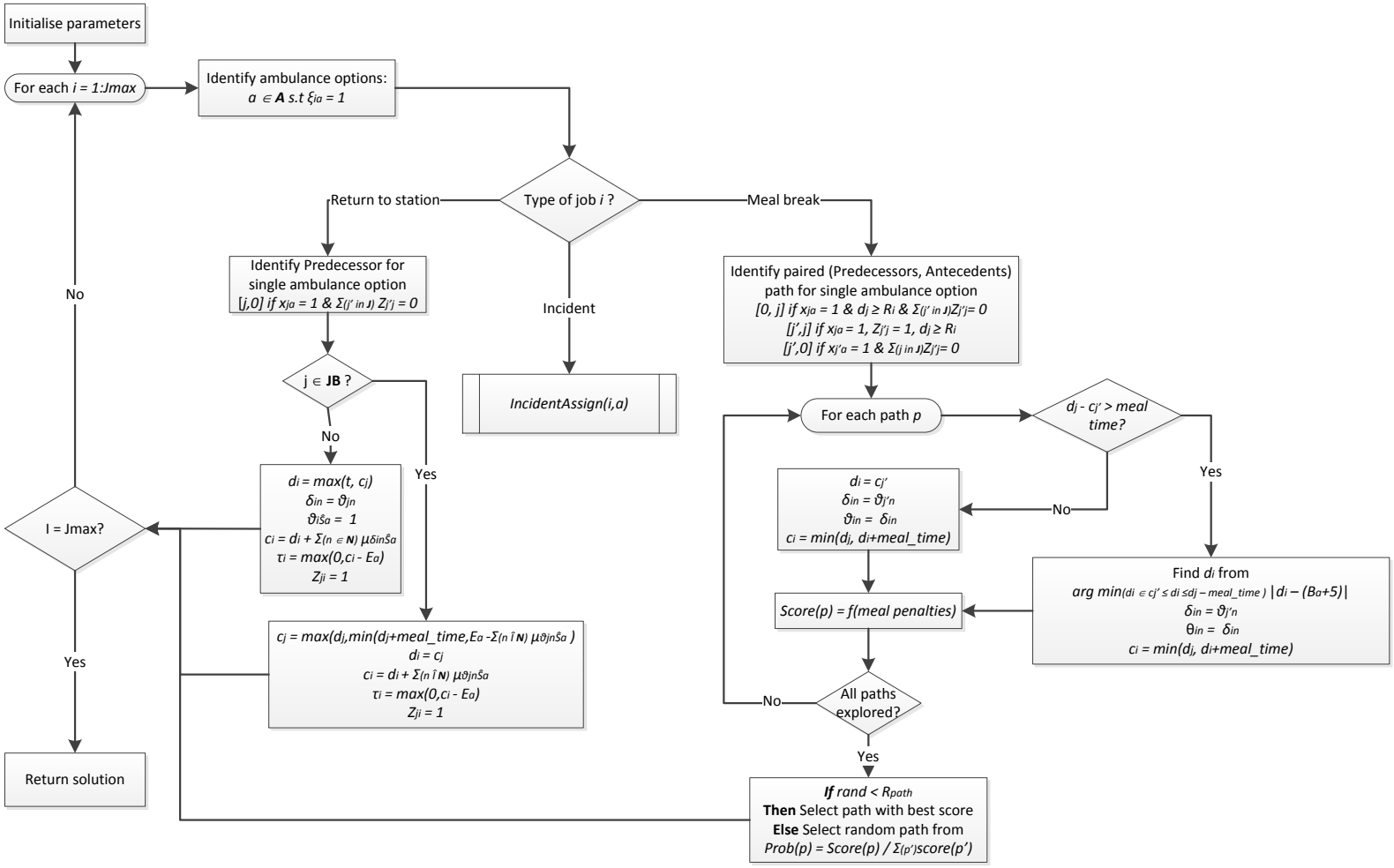


Figure 8-4 Part A of the process diagram for the CH to solve the real time model

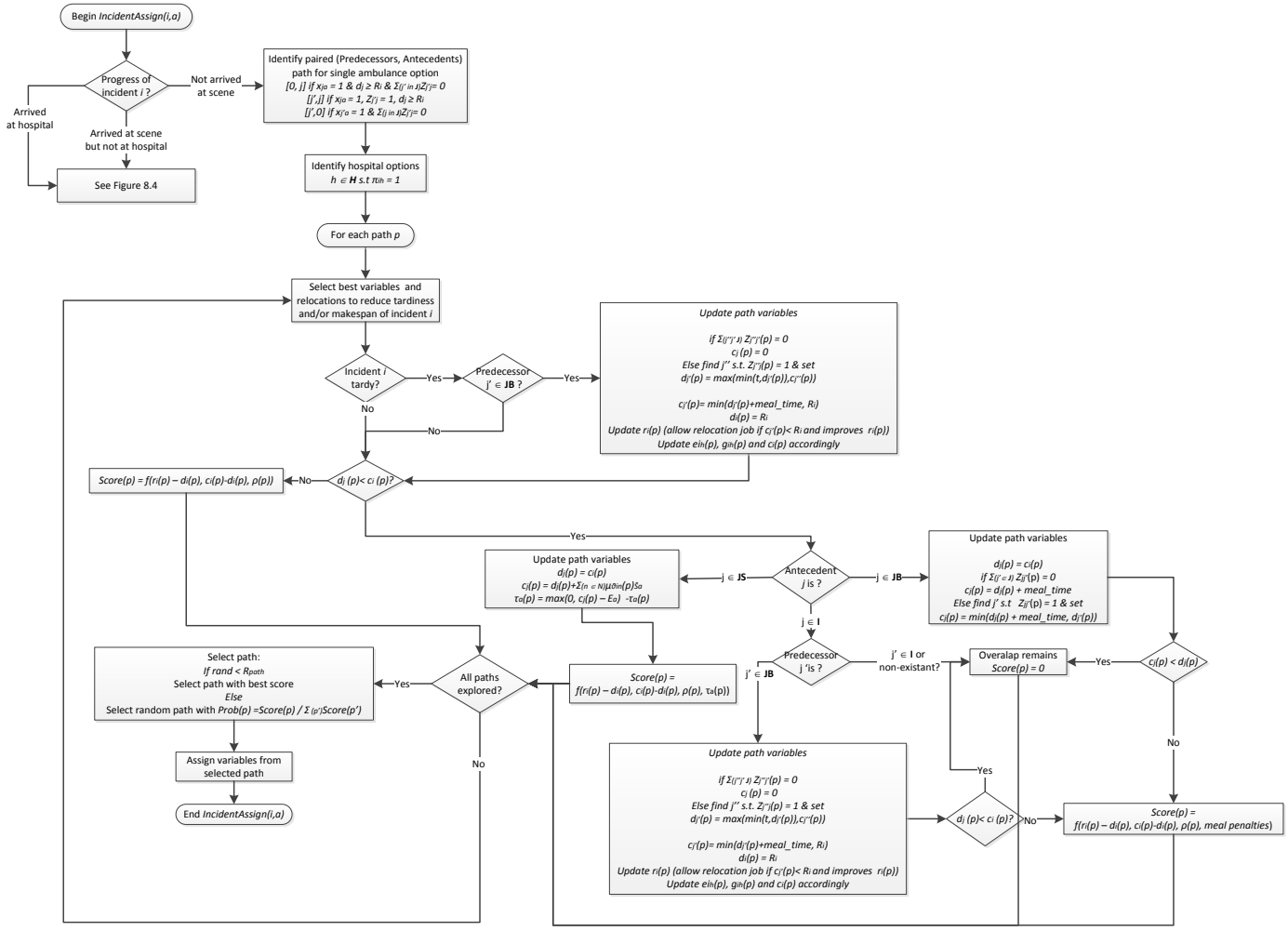


Figure 8-5 Part B of the process diagram for the CH to solve the real time model

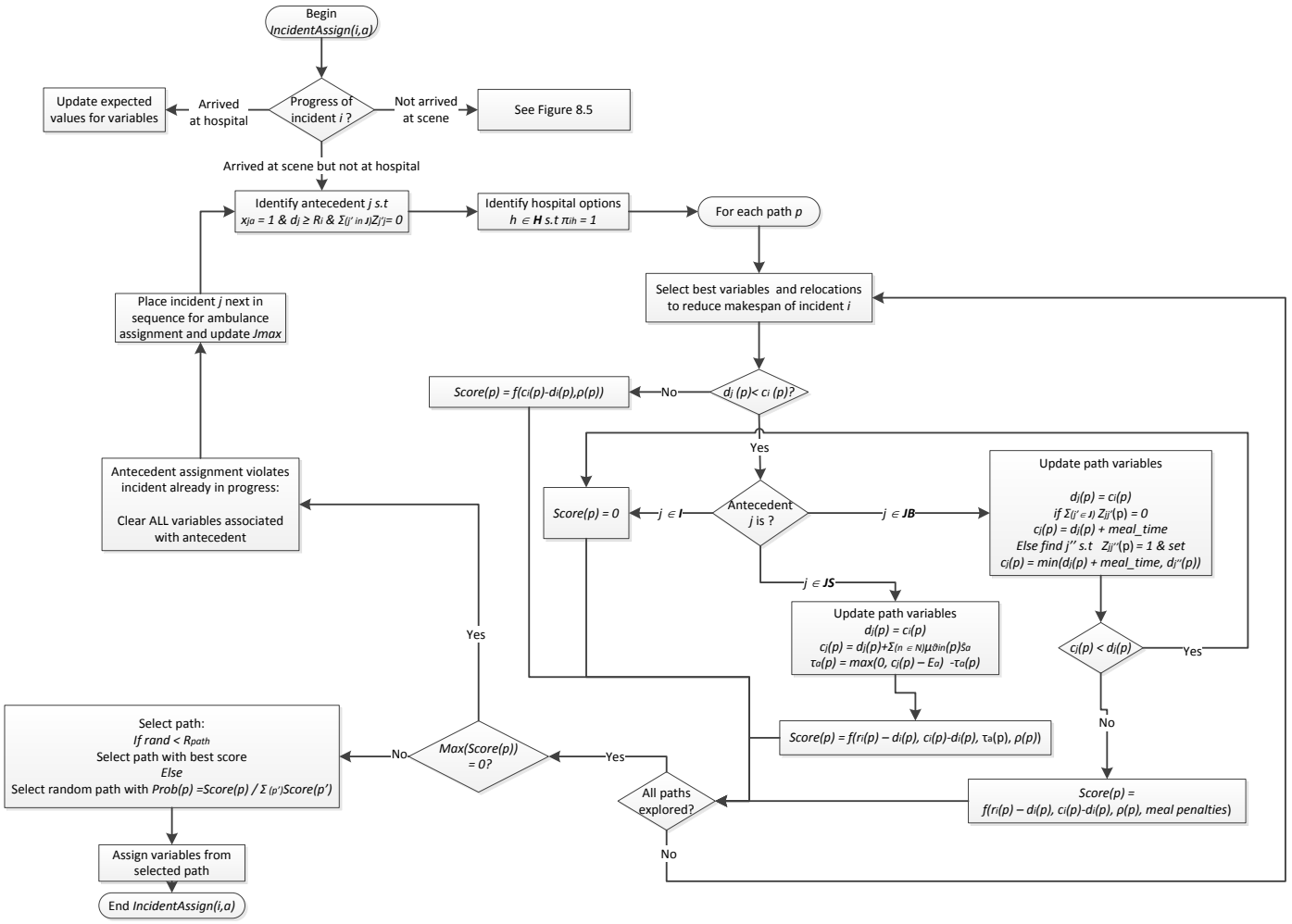


Figure 8-6 Part C of the process diagram for the CH to solve the real time model

Chapter 9: Conclusions

In this thesis, ambulance dispatching and relocation decisions have been modelled as a Flexible Flow Shop Scheduling Problem. Business rules for ambulance crew shift scheduling were integrated into the FFSS formulation as constraints. This allowed a single model to handle both ambulance scheduling and ambulance crew shift scheduling. Three models are presented in this thesis in Chapters 6, 7 and 8. Solutions to each model are analysed to compare solution approaches and verify the models.

Chapter 6 introduces the concept of the integrated ambulance scheduling and ambulance crew shift scheduling approach. It uses the simplifying assumption of static dispatching location to make the model easier to solve. Small case studies are able to be solved exactly. A larger case study is solved with heuristics. The proposed approach combined Tabu Search with a Constructive Heuristic. Results showed that response times for incidents requiring ambulance services can be reduced, by increasing the number of ambulances available. An upper bound is placed on the number of ambulance crew shifts, for heterogeneous ambulance types, that are required to meet response time targets in a case study.

Chapter 7 improves the model presented in Chapter 6. The simplifying assumption of static dispatching location is relaxed and additional ambulance crew shift scheduling rules are added into the model. Multiple solution heuristics are tested for this model, the most promising of which is a hybrid Ant Colony Optimisation and Constructive Heuristic. Results show an improved response time, with fewer ambulance crew shifts, than the previous model. This model can be solved strategically to build ambulance crew shift schedules from historical data. It can also be solved over short time periods (for example, hourly) to identify whether ambulances should be relocated and if additional ambulances may be required in the near future. A variation of this model is able to reduce response times for a fixed number of available ambulances.

Chapter 8 presents a model which uses real time data for the ambulance schedule problem. Fixed ambulance crew shift schedules and real time information

on ambulance locations, travelling times and incident form the input of this model. The resulting schedule contains ambulance dispatch, relocation decisions and meal and rest breaks for ambulance crew. A small case study is tested with an exact solver to investigate the quality of solutions with competing objectives. The model is effective when response time is included in the objective function but becomes difficult to solve when coverage is included.

9.1 RESPONSE TO RESEARCH AIMS

Five research aims, and eight research questions are proposed in Section 1.3.

The most utilised ambulance stations, and allocation of ambulances to each station, are investigated in the models in Chapter 6 and Chapter 7. It has been determined that, in terms of the number of ambulances scheduled to begin shifts at each location, there are some ambulance stations which are more utilised than others. The dynamic shift scheduling model, where minimising overtime is considered as part of the objective, tends toward scheduling a large number of ambulances at ambulance stations close to hospitals so as to minimise overtime at the end of the shift. Relocations are allowed immediately a shift begins and throughout a shift to ensure ambulance services are able to provide appropriate response times across an area.

The minimum number of ambulances is determined under static ambulance locations Chapter 6 and dynamic ambulance locations in Chapter 7. These models use performance requirements in the constraints to minimise ambulances while maintaining good response levels. The static model is able to provide an upper bound on the number of ambulances required, although this bound exceeds the number of ambulances scheduled in reality. The dynamic model reduces the number of ambulances required in the static model by allowing dynamic dispatch and relocations. It outperforms real response times and improves the upper bound on the number of ambulances required to meet performance requirements. This is potentially useful as a strategic planning tool which can be applied to investigate station locations and shift planning.

Algorithms to recommend dispatching decisions using a minimal number of crews are discussed in Chapters 7 and 8. The minimal number of ambulance crews as determined by the dynamic model, is used as input in a variation of the dynamic

model and the real time model. The variation of the dynamic model uses available data to schedule ambulance dispatching, relocation and return-to-station jobs. A case study, containing realistic information, is used to test this model. A hybrid Ant Colony Optimisation and Constructive Heuristic is able to build an ambulance schedule with very good response times. The real time model finds an ambulance schedule inclusive of dispatching decisions, relocations, breaks and return-station-jobs. A constructive heuristic is proposed for solving this model but requires real time data for further testing.

An ambulance crew scheduling methodology is developed under static and dynamic ambulance dispatching conditions. The method uses real business rules, such as maximum number of shifts per week, and implements them as constraints within a scheduling model. The ambulance crew schedule is validated by results showing that increasing the number of ambulance crews available each shift reduces the response times for all incidents.

This thesis investigates whether a mathematical model with minimum simplifying assumptions for real-life ambulance dispatch, scheduling and crew scheduling is necessary. The mathematical models developed use a scheduling approach which minimises assumptions about ambulance availability by directly assigning ambulances to demand. This approach is able to model overtime and include estimated hospital ramping times as a factor in the decision making process at the time of dispatch. This novel approach also uses expected overtime as a method of controlling the destinations to which ambulances can be relocated to improve coverage in the real time model, where other models use a travel time or distance approach which does not consider the amount of time remaining on shift. While overtime is not as large a contributor to the costs of running ambulance services as regular ambulance crew shifts, results show that overtime can be minimised without significant impact on the performance of ambulance services. A common approach in the literature for optimising ambulance locations is to maximise coverage. Results from the real time model in Chapter 8 demonstrate that this is a difficult objective to optimise in the scheduling model. Rather than assuming coverage requirements, the dynamic scheduling model uses expected incidents generated from historical data to schedule ambulances. This eliminates the requirement to have a complex model for estimating coverage requirements.

Multistage mathematical models in the literature assign ambulances to shifts in order to meet expected coverage requirements. Ambulance crew shifts are built around the incidents to which they respond. The integrated ambulance scheduling and ambulance crew scheduling models in Chapters 6 and 7 directly assign incidents to ambulances. The shift schedule for each ambulance crew is influenced by the features of incidents to which they are assigned. The benefit of the integrated model is the elimination of expected coverage determined by expected demand, in favour of using expected demand directly. This has resulted in good response times being achieved for all incidents, including non-urgent as well as emergency incidents. The drawback of this approach is the large number of decision variables present in the model, such that a rolling horizon approach with heuristic techniques becomes the most viable method of obtaining solutions.

Using a job shop scheduling approach, specifically flow shop scheduling, allows innovative use of disjunctive graphs to be applied to the integrated ambulance and ambulance crew scheduling problem. These allow the sequence of incidents to which ambulances respond to be used when building an ambulance crew schedule. This allows jobs to be introduced as the last to occur on each shift, which force ambulances to return to their home station from the location of the last job cleared. This is a useful approach for reassignment in the real time models and is a novel way to include overtime considerations. The limitation of this approach is the size of the disjunctive variables, which inflate the problem size significantly with each new incident.

The static model is the first to demonstrate the FFSS for integrated ambulance and ambulance crew scheduling. It is limited in use as a strategic planning tool and overestimates the number of ambulance crew requires. The dynamic model improves the static model. It is more realistic because it allows ambulances to be dispatched from any location rather than a single, static location. It allows ambulances to be relocated and can reassign ambulances to new incidents arriving to improve overall performance. The dynamic model can be solved as a strategic planning problem, with results reducing the response times and the number of ambulance crews required when compared to the static model for the same problem. However, the dynamic model has more variables than the static model, making problems larger and more difficult to solve.

The dynamic model and the real time model introduce the capability of solving the schedule in real time. This tactical approach offers several benefits over a strategic planning approach. Real time solutions allow the dispatching and relocation decisions made by ambulance dispatchers to be informed by a mathematical model. The dynamic model can also be solved over slightly longer time intervals, such as hourly intervals, using real time information as it becomes available to recommend ambulance relocations in the near future. These approaches have tested with small case studies. Further testing, with real time data, is still required.

The dynamic and real time models contain relocation and return-to-station jobs to model ambulance movements. This inflates the number of variables in each model where realistic problems are already large. Heuristic algorithms are necessary to solve the models in order to generate solutions in a reasonable amount of time. The quality of solutions for this approach is therefore affected by the quality of the solution heuristics. A Constructive Heuristic is able to find feasible solutions in seconds, suitable for an on-line solution. This study has demonstrated that hybrid heuristics, namely a hybrid Tabu Search and Constructive Heuristic, and a hybrid Ant Colony Optimisation and Constructive Heuristic, are able to find optimal solutions for small problem sizes. The ACO+CH algorithm is shown to perform well for larger problems with appropriately tuned parameters and a solution time of less than two minutes of CPU time for each hour of real time. Timely, on-line solutions for the dynamic and real time models require further testing and tuning of the heuristics with real time data.

There are anticipated benefits and costs for ambulance services implementing the scheduling models presented in this thesis. Costs include: potentially increasing ambulance crew numbers in order to attain improvements to response times; a greater number of ambulance relocations each shift; and changes in the number of ambulances assigned to each ambulance station. Anticipated benefits to ambulance services are: reduced response times for emergency incidents; reduced response times for all incidents; and reduced overtime for ambulance crews.

9.2 COMPARISON OF EACH MODEL

The limitations and benefits of each of the three models are presented here. Table 9-1 compares the limitations for the static, dynamic and reactive models

while Table 9-2 compares the benefits. In essence, the static model is a simple, proof of concept model, with a large number of assumptions but able to be solved more easily than the other models. The dynamic model outperforms the static model and has fewer simplifying assumptions, therefore, it has fewer limitations and more benefits. It also introduces the concept of relocation and return to station events as jobs to be scheduled and the novel disjunctive location variables to track ambulance locations, making it possible to model relocations and consider overtime in a single model. However, the dynamic model is complex and has a large number of variables. It uses more complicated solution algorithms than the static model. The reactive model is different to the other models. It does not build a crew schedule and cannot, at this point, add additional ambulances into the system. It uses real data as it becomes available and an estimation of coverage requirements to aid in the making of relocation decisions. The reactive model is able to schedule meal and rest breaks around incident responses and relocations. It also relaxes the constraints in the static and dynamic models limiting response times, allowing response time to be an objective criterion. The reactive model requires further testing.

Table 9-1 Limitations of each model presented in this thesis

Model 1: Static	**Model 2: Dynamic**	**Model 3: Reactive**
Uses deterministic data	Tested with deterministic data	Requires pre-defined crew schedules and cannot call additional resources
Tested with small case study: only schedules one week of data and does not allow ambulances to enter or leave the region	Tested with a number of small case studies: only schedule one week of data at a time and do not allow ambulances to enter or leave the region	Requires estimation of coverage requirements in advance
Does not allow relocation of ambulances	Complex model: Lots of decision variables and disjunctive variables	Requires testing of appropriate look ahead time
Does not allow pre-emption at any time	More complex heuristic algorithms needed to solve realistic size problems	Requires exploration of appropriate weights for objective criteria

Does not schedule meal or rest breaks	Does not schedule meal or rest breaks	Assumes meal and rest breaks can be taken at any location
No limit on number of shifts per week or number of consecutive night shifts	Assumes all ambulance types can transfer to hospital	No limitation on the number of incidents which can be tardy
Ambulances must return to home station before being dispatched again, each ambulance only able to be dispatched from one location	Assumes three type of ambulances, with incidents requesting a particular type also able to be served by more costly ambulance types	No upper bound on the response time
Models incidents not required to go to hospital as requiring transfer to a dummy hospital	Ramping time independent of number of patients already sent to hospital	Assumes all ambulance types can transfer to hospital
Assumes all ambulance types can transfer to hospital	Assumes ambulance vehicles staffed by fixed ambulance crews working the same schedule together	Assumes three type of ambulances, with incidents requesting a particular type also able to be served by more costly ambulance types
Assumes three type of ambulances, with incidents requesting a particular type also able to be served by more costly ambulance types		Ramping time independent of number of patients already sent to hospital
Ramping time independent of number of patients already sent to hospital		Assumes ambulance vehicles staffed by fixed ambulance crews working the same schedule together
Assumes ambulance vehicles staffed by fixed ambulance crews working the same schedule together		Must be solved quickly to be of use
Known to overestimate the number of ambulances required per shift		

Table 9-2 Benefits of each model presented in this thesis

Static model	Dynamic model	Reactive model
Returns an ambulance crew schedule to meet performance targets with minimal costs	Returns an ambulance crew schedule to meet performance targets with minimal costs	Formulated to be solved reactively, using information as it becomes known
Simple model that acts as proof of concept for scheduling techniques as a way to integrate ambulance scheduling and ambulance crew scheduling	Minimises costs more accurately than static model	Uses expected coverage at look ahead time, and estimated overtime if travel to home ambulance station will be required, to make relocation decisions
Ensures all incidents will receive a response within a given time window	Able to be solved for rolling time horizons, where information can be updated each horizon	Schedules meal and rest breaks, and allowes these to be interrupted for penalty costs
Ensures ambulances will always return to their home station at the end of a shift	Allows relocation of ambulances based on expected demand	Ambulances can still be dispatched to jobs even after the shift was due to end
Allows multiple ambulance types	Allows pre-emption during relocation, initial response phase and when travelling to hospital	Ensures ambulances will always return to their home station at the end of a shift
Allows hospital selection	Includes limits on shifts per week, consecutive night shifts and fatigue breaks	Allows multiple ambulance types
Limits the number of tardy responses	Ensures all incidents will receive a response within a given time window	Allows hospital selection
Small problem solvable with CPLEX, larger problem solvable with TS+CH algorithm	Ensures ambulances will always return to their home station at the end of a shift	
	Allows multiple ambulance types	
	Allows hospital selection	
	Limits the number of tardy responses	

	Can be adapted to use crew schedules as parameters and minimise response times	
	Tracks incident status and ambulance locations	
	Introduces novel disjunctive location variables	
	Promising results from TS+CH and ACO+CH algorithms	

9.3 FUTURE RESEARCH DIRECTIONS

Several avenues for future work have been identified in this thesis. These include extending the case study, improving the heuristics and reducing the simplifying assumptions used to develop the mathematical models.

The case study can be extended to cover a larger area which includes additional ambulance stations and hospitals, as well as more incidents. This will make the problem harder to solve but will better model the cooperation between ambulances and hospitals in a metropolitan region. Locations at which stations do not currently exist can also be introduced into the model to test whether schedules can be improved by establishing new ambulance stations. The case study can also be developed to cover a longer period of time, allowing crew schedules to be built for several weeks at a time. Using several weeks of data to build crew schedules will also allow the robustness of the results to be investigated. Improvements to the processing times in the case study can be made through calling time dependent travel times from a road network, which will provide more accurate data for the scheduling models. Distributions for ramping time and time spent on scene may also be improved. Coverage requirements used in the real time model may also be improved through application of double coverage models and hypercube models from the literature.

Identified options for improving the heuristics include modifying the hybrid ACO+CH to vary parameters according to the size of the problem. Analysis of the heuristic for the dynamic model showed that better solutions for smaller problems

were found with a larger number of ants. The hybrid TS+CH may also be improved through additional sensitivity analysis of the weights on performance measures used in the smart swap algorithm or through testing a different set of performance measures. Finally, application of a hyper heuristic using ACO+CH to vary the sequence of incidents initially, followed by running the TS+CH methodology to refine the sequence, may offer improvements to the solution.

Assumptions used in the scheduling models which it is desirable to relax in future work are: shift patterns, hospital preference assumptions, ambulance vehicle type requirements and fixed schedules in the real time model. Shift patterns have been restricted to three shift options per day to reduce the size of the problems being solved. The number of shifts per day can be increased. This will make the problem harder to solve due to the increase in decision variables, but is expected to reduce the number of ambulances required to meet demand and provide advice to strategic planners on possible crew schedules. Crew schedules in the real time model do not yet allow the option for dispatchers, faced with higher than expected demand, to request more ambulances to begin work. This is identified as an important extension to integrate ambulance and ambulance crew scheduling in order to select optimal crew to call in for additional hours.

Bibliography

Ak, B., & Koc, E. (2012). A guide for genetic algorithm based on parallel machine scheduling and flexible job-shop scheduling. *Procedia - Social and Behavioral Sciences, 62*(0), 817-823.

Almehdawe, E., Jewkes, B., & He, Q.-M. (2013). A Markovian queueing model for ambulance offload delays. *European Journal of Operational Research, 226*(3), 602-614.

Amiri, M., Zandieh, M., Yazdani, M., & Bagheri, A. (2010). A variable neighbourhood search algorithm for the flexible job-shop scheduling problem. *International Journal of Production Research, 48*(19), 5671-5689.

Andersson, T., & Värbrand, P. (2007). Decision support tools for ambulance dispatch and relocation. *The Journal of the Operational Research Society, 58*(2), 195-201.

Aubin, J. (1992). Scheduling Ambulances. *Interfaces, 22*(2), 1-10.

Azadeh, A., Hosseinabadi Farahani, M., Torabzadeh, S., & Baghersad, M. (2014). Scheduling prioritized patients in emergency department laboratories. *Computer Methods and Programs in Biomedicine, 117*(2), 61-70.

Bagheri, A., Zandieh, M., Mahdavi, I., & Yazdani, M. (2010). An artificial immune algorithm for the flexible job-shop scheduling problem. *Future Generation Computer Systems, 26*(4), 533-541.

Beaudry, A., Laporte, G., Melo, T., & Nickel, S. (2009). Dynamic transportation of patients in hospitals. *OR Spectrum, 32*(1), 77-107.

Becker, T. K., Gausche-Hill, M., Aswegan, A. L., Baker, E. F., Bookman, K. J., Bradley, R. N., . . . American College of Emergency Physicians, E. M. S. C. (2013). Ethical challenges in Emergency Medical Services: controversies and recommendations. *Prehospital and disaster medicine, 28*(5), 488-497.

Beraldi, P., & Bruni, M. E. (2009). A probabilistic model applied to emergency service vehicle location. *European Journal of Operational Research, 196*(1), 323-331.

Błażewicz, J., Pesch, E., & Sterna, M. (2000). The disjunctive graph machine representation of the job shop scheduling problem. *European Journal of Operational Research, 127*(2), 317-331.

Brandimarte, P. (1993). Routing and scheduling in a flexible job shop by tabu search. *Annals of Operations Research, 41*(3), 157-183.

Brotcorne, L., Laporte, G., & Semet, F. (2003). Ambulance location and relocation models. *European Journal of Operational Research, 147*(3), 451-463.

Brucker, P., Jurisch, B., & Krämer, A. (1997). Complexity of scheduling problems with multi-purpose machines. *Annals of Operations Research, 70*, 57-73.

Brucker, P., & Knust, S. (2006). *Complex Scheduling*. Berlin: Springer-Verlag.

Burke, E. K., Kendall, G., & Soubeiga, E. (2003). A tabu-search hyperheuristic for timetabling and rostering. *Journal of Heuristics, 9*(6), 451-470.

Burke, E. K., McCollum, B., Meisels, A., Petrovic, S., & Qu, R. (2007). A graph-based hyper-heuristic for educational timetabling problems. *European Journal of Operational Research, 176*(1), 177-192.

Church, R., & ReVelle, C. (1974). The maximal covering location problem. *Papers in Regional Science, 32*(1), 101-118.

Church, R., Sorensen, P., & Corrigan, W. (2001). Manpower deployment in emergency services. *Fire Technology, 37*(3), 219-234.

Corry, P., & Kozan, E. (2004). Ant colony optimisation for machine layout problems. *Computational Optimization and Applications, 28*(3), 287-310.

Daskin, M. S. (1982). Application of an expected covering model to emergency medical service system design. *Decision Sciences, 13*(3), 416-439.

Dauzère-Pérès, S., & Paulli, J. (1997). An integrated approach for modeling and solving the general multiprocessor job-shop scheduling problem using tabu search. *Annals of Operations Research, 70*(0), 281-306.

Dorigo, M., & Blum, C. (2005). Ant colony optimization theory: A survey. *Theoretical Computer Science, 344*(2–3), 243-278.

Erdogan, G., Erkut, E., Ingolfsson, A., & Laporte, G. (2010). Scheduling ambulance crews for maximum coverage. *Journal of the Operations Research Society, 61*(4), 543-550.

Ernst, A. T., Jiang, H., Krishnamoorthy, M., Owens, B., & Sier, D. (2004). An annotated bibliography of personnel scheduling and rostering. *Annals of Operations Research, 127*(1), 21-144.

Ernst, A. T., Jiang, H., Krishnamoorthy, M., & Sier, D. (2004). Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research, 153*(1), 3-27.

French, S. (1982). *Sequencing and scheduling : An introduction to the mathematics of the job-shop*. Chichester, West Sussex: E. Horwood.

Gao, J., Sun, L., & Gen, M. (2008). A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems. *Computers & Operations Research, 35*(9), 2892-2907.

García-Villoria, A., Salhi, S., Corominas, A., & Pastor, R. (2011). Hyper-heuristic approaches for the response time variability problem. *European Journal of Operational Research, 211*(1), 160-169.

Gendreau, M., Laporte, G., & Semet, F. (1997). Solving an ambulance location model by tabu search. *Location Science, 5*(2), 75-88.

Gendreau, M., Laporte, G., & Semet, F. (2001). A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Computing, 27*(12), 1641-1653.

Gendreau, M., Laporte, G., & Semet, F. (2006). The maximal expected coverage relocation problem for emergency vehicles. *The Journal of the Operational Research Society, 57*(1), 22-28.

Gendreau, M., & Potvin, J.-Y. (2005). Chapter 6: Tabu Search. In E. K. Burke & G. Kendall (Eds.), *Search Methodologies - Introductory tutorials in optimization and decision support techniques* (pp. 22-28). New York: Springer.

Gendreau, M., & Potvin, J.-Y. (2010). Tabu Search. In M. Gendreau & J.-Y. Potvin (Eds.), *Handbook of Metaheuristics* (Vol. 146, pp. 41-59). New York: Springer.

Geroliminis, N., Karlaftis, M. G., & Skabardonis, A. (2009). A spatial queuing model for the emergency vehicle districting and location problem. *Transportation Research Part B: Methodological, 43*(7), 798-811.

Girish, B. S., & Jawahar, N. (22-25 Aug. 2009). A particle swarm optimization algorithm for flexible job shop scheduling problem. In *IEEE International Conference on Automation Science and Engineering, CASE 2009* (pp. 298-303), Piscataway, NJ, USA, IEEE.

Glover, F. W., & Laguna, M. (1997). *Tabu Search*. New York: Springer.

Goldberg, J. B. (2004). Operations research models for the deployment of emergency services vehicles. *EMS Management Journal, 1*(1), 20-39.

Goodwin, G. C., & Medioli, A. M. (2013). Scenario-based, closed-loop model predictive control with application to emergency vehicle scheduling. *International Journal of Control, 86*(8), 1338-1348.

Haghani, A., Tian, Q., & Hu, H. (2004). Simulation model for real-time emergency vehicle dispatching and routing. *Transportation Research Record: Journal of the Transportation Research Board, 1882*(-1), 176-183.

Haghani, A., & Yang, S. (2007). Real-time emergency response fleet deployment: Concepts, systems, simulation & case studies. In V. Zeimpekis, C. D. Tarantilis, G. M. Giaglis & I. Minis (Eds.), *Dynamic Fleet Management : Concepts, Systems, Algorithms and Case Studies* (pp. 133-162). Dordrecht: Springer.

Hansen, P., Mladenović, N., Brimberg, J., & Pérez, J. M. (2010). Variable Neighborhood Search. In M. Gendreau & J.-Y. Potvin (Eds.), *Handbook of Metaheuristics* (Vol. 146, pp. 61-86). New York: Springer.

Henderson, D., Jacobson, S., & Johnson, A. (2003). The theory and practice of simulated annealing. In F. Glover & G. Kochenberger (Eds.), *Handbook of Metaheuristics* (Vol. 57, pp. 287-319). New York: Springer.

Henderson, S., & Mason, A. (2005). Ambulance service planning: Simulation and data visualisation. In M. Brandeau, F. Sainfort & W. Pierskalla (Eds.), *Operations Research and Health Care* (Vol. 70, pp. 77-102). New York: Springer.

Henderson, S. G., & Mason, A. J. (1999). Estimating ambulance requirements in Auckland, New Zealand. In P. Farrington, H. Nembhard, J. Evans & D. Sturrock (Eds.), *Winter Simulation Conference Proceedings* (Vol. 2, pp. 1670-1674). Piscataway, New Jersey: IEEE.

Hurink, J., Jurisch, B., & Thole, M. (1994). Tabu search for the job-shop scheduling problem with multi-purpose machines. *OR Spectrum, 15*(4), 205-215.

Iannoni, A. P., & Morabito, R. (2007). A multiple dispatch and partial backup hypercube queuing model to analyze emergency medical systems on highways. *Transportation Research Part E: Logistics and Transportation Review, 43*(6), 755-771.

Ibri, S., Drias, H., & Nourelfath, M. (2010). Integrated emergency vehicle dispatching and covering: A parallel Ant-tabu approach. In *Proceedings of the 8th International Conference of Modeling and Simulation, MOSIM'10*, 1039-1045.

Ingolfsson, A., Budge, S., & Erkut, E. (2008). Optimal ambulance location with random delays and travel times. *Health Care Management Science, 11*(3), 262-274.

Jianga, J., Wena, M., Maa, K., Longa, X., & Lia, J. (2011). Hybrid genetic algorithm for flexible job-shop scheduling with multi-objective. *Journal of Information and Computational Science, 8*, 2197-2205.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization, In *Proceedings of IEEE international conference on neural networks* 1942-1948, Piscataway, NJ, IEEE.

Kergosien, Y., Lenté, C., Piton, D., & Billaut, J. C. (2011). A tabu search heuristic for the dynamic transportation of patients between care units. *European Journal of Operational Research, 214*(2), 442-452.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*(4598), 671-680.

Kozan, E., & Mesken, N. (2005). A simulation model for emergency centres. In A. Zerger & R. Argent (Eds.), MODSIM 2005 *International Congress on Modelling and Simulation.* Modelling and Simulation Society of Australia and New Zealand, December 2005, pp. 2602-2608. ISBN: 0-9758400-2-9. www.mssanz.org.au/modsim05/papers/kozan.pdf

Larson, R. C. (1974). A hypercube queuing model for facility location and redistricting in urban emergency services. *Computers & Operations Research, 1*(1), 67-95.

Li, X., Zhao, Z., Zhu, X., & Wyatt, T. (2011). Covering models and optimization techniques for emergency response facility location and planning: A review. *Mathematical Methods of Operations Research, 74*(3), 281-310.

Li, Y., & Kozan, E. (2009). Rostering ambulance services. In *Asia Pacific Industrial Engineering and Management Systems Conference* (pp. 795-801), Kitakyushu, Japan.

Maleki, M., Majlesinasab, N., & Sepehri, M. M. (2014). Two new models for redeployment of ambulances. *Computers & Industrial Engineering, 78*(0), 271-284.

Mason, A. (2005). Emergency vehicle trip analysis using GPS AVL data: A dynamic program for map matching. In *Proceedings of the 40th Annual Conference of the Operational Research Society of New Zealand* (pp. 295-304), *Wellington, New Zealand*

Mati, Y., & Xie, X. (2004). The complexity of two-job shop problems with multi-purpose unrelated machines. *European Journal of Operational Research, 152*(1), 159-169.

Maxwell, M. S., Ni, E. C., Tong, C., Henderson, S. G., Topaloglu, H., & Hunter, S. R. (2014). A bound on the performance of an optimal ambulance redeployment policy. *Operations Research, 62*(5), 1014-1027.

Maxwell, M. S., Restrepo, M., Henderson, S. G., & Topaloglu, H. (2010). Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing, 22*(2), 266-281.

McLay, L., & Mayorga, M. (2010). Evaluating emergency medical service performance measures. *Health Care Management Science, 13*(2), 124-136.

Melachrinoudis, E., Ilhan, A. B., & Min, H. (2007). A dial-a-ride problem for client transportation in a health-care organization. *Computers and Operations Research, 34*(3), 742-759.

Melachrinoudis, E., & Min, H. (2011). A tabu search heuristic for solving the multi-depot, multi-vehicle, double request dial-a-ride problem faced by a healthcare organisation. *International Journal of Operational Research, 10*(2), 214-239.

Mendonça, F. C., & Morabito, R. (2001). Analysing emergency medical service ambulance deployment on a Brazilian highway using the hypercube model. *The Journal of the Operational Research Society, 52*(3), 261-270.

Mesghouni, K., Hammadi, S., & Borne, P. (2004). Evolutionary algorithms for job-shop scheduling. *International Journal of Applied Mathematics and Computer Science, 14*(1), 91-104.

Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research, 24*(11), 1097-1100.

Moslehi, G., & Mahnam, M. (2011). A Pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search. *International Journal of Production Economics, 129*(1), 14-22.

Nie, L., Gao, L., Li, P., & Li, X. (2012). A GEP-based reactive scheduling policies constructing approach for dynamic flexible job shop scheduling problem with job release dates. *Journal of Intelligent Manufacturing*, 1-12.

Öncan, T. (2007). A survey of the generalized assignment problem and its applications. *INFOR: Information Systems and Operational Research, 45*(3), 123-141.

Parragh, S. (2009). *Ambulance routing problems with rich constraints and multiple objectives* (Doctoral dissertation). uniwien. Retrieved 4 February 2015 from http://othes.univie.ac.at/5380/

Pezzella, F., Morganti, G., & Ciaschetti, G. (2008). A genetic algorithm for the Flexible Job-shop Scheduling Problem. *Computers & Operations Research, 35*(10), 3202-3212.

Pillay, N., & Banzhaf, W. (2009). A study of heuristic combinations for hyper-heuristic systems for the uncapacitated examination timetabling problem. *European Journal of Operational Research, 197*(2), 482-491.

Pinedo, M. L. (2012). *Scheduling: Theory, Algorithms, and Systems.* (4th ed.). New York: Springer.

Pitts, R. A., & Ventura, J. A. (2009). Scheduling flexible manufacturing cells using Tabu Search. *International Journal of Production Research, 47*(24), 6907-6928.

Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm Intelligence, 1*(1), 33-57.

Queensland Ambulance Service. (2014). QAS public webpage. Retrieved 1 August 2014, from Department of Community Safety, www.ambulance.qld.gov.au

Queensland Ambulance Service. (2014). Queensland Ambulance Service Publications. Retrieved 10 September 2014, from Queensland Government, https://ambulance.qld.gov.au/publications.html

Queensland Department of Community Safety. (2012). *Annual Report 2011-12*. Retrieved 23 January 2013, from http://www.communitysafety.qld.gov.au/Annual_reports/Annual_Report_2011-2012/default.htm

Queensland Health. (2013). Hospital Performance. Retrieved 14 June 2013 from http://www.health.qld.gov.au/hospitalperformance/default.aspx

Queensland Industrial Relations Commission. (2012). *Ambulance Service Employees' Award*. Retrieved 3 February 2015 from http://www.qirc.qld.gov.au/

Queensland Treasury. (2007). *Queensland Ambulance Service Audit Report*. http://www.emergency.qld.gov.au/publications/pdf/FinalReport.pdf

Rajabinasab, A., & Mansour, S. (2011). Dynamic flexible job shop scheduling with alternative process plans: an agent-based approach. *The International Journal of Advanced Manufacturing Technology, 54*(9-12), 1091-1107.

Rajagopalan, H. K., Saydam, C., Sharer, E., & Setzler, H. (2011). Ambulance deployment and shift scheduling: An integrated approach. *Journal of Service Science and Management, 4*(1), 66-78.

Rajagopalan, H. K., Saydam, C., & Xiao, J. (2008). A multiperiod set covering location model for dynamic redeployment of ambulances. *Computers & Operations Research, 35*(3), 814-826.

Reeves, C. (2003). Genetic Algorithms. In F. Glover & G. Kochenberger (Eds.), *Handbook of Metaheuristics* (Vol. 57, pp. 55-82). New York: Springer.

Repede, J. F., & Bernardo, J. J. (1994). Developing and validating a decision support system for locating emergency medical vehicles in Louisville, Kentucky. *European Journal of Operational Research, 75*(3), 567-581.

Rosengren, D. (2012). *A report on ambulance ramping in metropolitan hospitals*. http://www.health.qld.gov.au/publications/medai-report/final_medai_report.pdf

Saidi-Mehrabad, M., & Fattahi, P. (2007). Flexible job shop scheduling with tabu search algorithms. *The International Journal of Advanced Manufacturing Technology, 32*(5), 563-570.

Schmid, V., & Doerner, K. F. (2010). Ambulance location and relocation problems with time-dependent travel times. *European Journal of Operational Research, 207*(3), 1293-1303.

Shanker, K., & Modi, B. K. (1999). A branch and bound based heuristic for multi-product resource constrained scheduling problem in FMS environment. *European Journal of Operational Research, 113*(1), 80-90.

Taha, H. A. (2003). *Operations research: An introduction*. Upper Saddle River, NJ: Pearson Education.

Tay, J. C., & Ho, N. B. (2008). Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems. *Computers & Industrial Engineering, 54*(3), 453-473.

Toregas, C., Swain, R., ReVelle, C., & Bergman, L. (1971). The location of emergency service facilities. *Operations Research, 19*(6), 1363-1373.

Trudeau, P., Rousseau, J.-M., Ferland, J. A., & Choquette, J. (1989). An operations research approach for the planning and operation of an ambulance service. *INFOR, 27*(1), 95-113.

Wang, H. (2005). Flexible flow shop scheduling: Optimum, heuristics and artificial intelligence solutions. *Expert Systems, 22*(2), 78-85.

Westgate, B. S., Woodard, D. B., Matteson, D. S., & Henderson, S. G. (2013). Travel time estimation for ambulances using Bayesian data augmentation. *The Annals of Applied Statistics, 7*(2), 1139-1161.

Wilson, D. T., Hawe, G. I., Coates, G., & Crouch, R. S. (2013). A multi-objective combinatorial model of casualty processing in major incident response. *European Journal of Operational Research*.

Xia, W., & Wu, Z. (2005). An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems. *Computers & Industrial Engineering, 48*(2), 409-425.

Xiang, W., & Lee, H. P. (2008). Ant colony intelligence in multi-agent dynamic manufacturing scheduling. *Engineering Applications of Artificial Intelligence, 21*(1), 73-85.

Xiang, W., Yin, J., & Lim, G. (2014). A short-term operating room surgery scheduling problem integrating multiple nurses roster constraints. [Accepted Proof]. *Artificial Intelligence in Medicine*.

Xing, L. N., Chen, Y. W., Wang, P., Zhao, Q. S., & Xiong, J. (2010). Knowledge-based ant colony optimization for flexible job shop scheduling problems. *Applied Soft Computing, 10*(3), 888-896.

Yang, S., Hamedi, M., & Haghani, A. (2005). Online dispatching and routing model for emergency vehicles with area coverage constraints. *Transportation Research Record: Journal of the Transportation Research Board, 1923*, 1-8.

Yang, X.-S. (2009). Harmony search as a metaheuristic algorithm. In Z. Geem (Ed.), *Music-Inspired Harmony Search Algorithm* (Vol. 191, pp. 1-14). Berlin: Springer.

Yi, W., & Kumar, A. (2007). Ant colony optimization for disaster relief operations. *Transportation Research Part E: Logistics and Transportation Review, 43*(6), 660-672.

Zakaria, Z., & Petrovic, S. (2012). Genetic algorithms for match-up rescheduling of the flexible manufacturing systems. *Computers & Industrial Engineering, 62*(2), 670-686.

Zhang, G., Shao, X., Li, P., & Gao, L. (2009). An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem. *Computers & Industrial Engineering, 56*(4), 1309-1318.

Zhang, L. (2012). *Simulation optimisation and Markov models for dynamic ambulance redeployment* (Doctoral dissertation, Engineering Science). University of Auckland. Retrieved 4 February 2015 from https://researchspace.auckland.ac.nz/handle/2292/20319

Zhen, L., Sheng, S., Xie, Z., & Wang, K. (2014). Decision rules for ambulance scheduling decision support systems. *Applied Soft Computing Journal, 26*, 350-356.

Zhou, R., Nee, A. Y. C., & Lee, H. P. (2009). Performance of an ant colony optimisation algorithm in dynamic job shop scheduling problems. *International Journal of Production Research, 47*(11), 2903-2920.

Zong Woo Geem, Joong Hoon Kim, & Loganathan, G. V. (2001). A new heuristic optimization algorithm: Harmony search. *Simulation, 76*(2), 60-68.